

---

---

# HiBeauty! Recomendaciones para la piel

---

---

Por  
Carmen Serrano Avilés



UNIVERSIDAD COMPLUTENSE  
MADRID

Trabajo de fin de grado del Grado en Ingeniería Informática  
FACULTAD DE INFORMÁTICA

*Dirigido por*  
Ana María González de Miguel  
**HiBeauty Recommendations for the skin**

MADRID, 2020–2021

HiBeauty! Recomendaciones para la piel

# Hi Beauty!

Una aplicación para el cuidado de la piel

*Memoria que se presenta para el Trabajo de Fin de Grado*

**Carmen Serrano Avilés**

*Dirigido por*

**Ana María González de Miguel**

**Departamento de Ingeniería del Software e Inteligencia Artificial**

**Facultad de Informática**

**Universidad Complutense de Madrid**

**Madrid, 2020**



# Agradecimientos

En primer lugar y antes de empezar con la explicación del desarrollo de la aplicación quiero añadir algunos agradecimientos.

A Ana María González de Miguel, porque sin ella este proyecto no hubiera salido adelante, quiero agradecerle cada reunión y cada una de las palabras, minutos y segundos que me ha dedicado a lo largo de este curso. Gracias por confiar en mi y en mi idea, no lo olvidaré. Trabajar y consolidar las ideas contigo ha sido todo un acierto y creo que no hubiera podido hacerlo mejor con otra persona que no seas tú.

A los profesores e integrantes de la Facultad de Informática por aportar el granito de conocimiento que me ha ayudado a evolucionar de una manera académica, personal y en futuro, profesional.

A mi familia, pareja y amigos, por apoyarme durante este camino tan largo pero gratificante como han sido los años de la carrera y el desarrollo de este proyecto.

# Abstracto

Desde hace algunos años, el cuidado de la piel del rostro está en auge, lo que implica que cada vez más usuarios se preocupen por el mismo.

*HiBeauty!* es una aplicación para ayudar a dichos usuarios que quieran cuidar de una manera distinta y mejor su rostro.

Este proyecto recomienda productos en función de la piel que especifica el usuario, muestra los ingredientes dañinos o nocivos para que el usuario decida si quiere o no comprar o seguir utilizando el producto que ha escaneado previamente para su análisis.

Este desarrollo se puede ver implementado tanto en una página web como en una apk para teléfonos móviles.

**Palabras clave** Productos, cosméticos, EWG, ingredientes, OCR.

# Abstract

For some years now, facial skin care has been on the rise, which means that more and more users are worrying about it.

*HiBeauty!* is an application to help those users who want to take better care of their facial skin.

This project recommends products based on the skin that the user specifies, shows the harmful or toxic ingredients so that the user decides whether or not to buy or continue using the product that he has previously scanned for analysis.

This development can be seen implemented both on a web page and in an apk for mobile phones.

**Keywords** Products, cosmetics, EWG, ingredients, OCR.

# Índice general

	Página
<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	2
1.2. Objetivos . . . . .	2
1.3. Plan de Trabajo . . . . .	3
1.4. Conclusiones del capítulo . . . . .	4
<b>2. Introduction</b>	<b>1</b>
2.1. Motivation . . . . .	1
2.2. Objectives . . . . .	2
2.3. Work-plan . . . . .	3
2.4. Chapter conclusion . . . . .	3
<b>3. Estado del arte</b>	<b>5</b>
3.1. Aplicaciones similares . . . . .	5
3.1.1. Clean Beauty . . . . .	5
3.1.2. 1Source . . . . .	6
3.2. Aplicaciones web similares . . . . .	7
3.2.1. Inci . . . . .	7
3.3. Tecnologías de OCR . . . . .	8
3.3.1. ¿Qué es el OCR? . . . . .	8
3.3.2. Tecnologías de OCR descartadas . . . . .	8
3.3.3. Tecnología OCR utilizada . . . . .	9
3.4. Bases de Datos . . . . .	10
3.5. Conclusiones del capítulo . . . . .	11



<b>4. Escenarios y requisitos</b>	<b>12</b>
4.1. Escenarios . . . . .	13
4.2. Requisitos funcionales . . . . .	14
4.2.1. Productos . . . . .	14
4.2.2. Cámara . . . . .	15
4.2.3. Usuario . . . . .	15
4.3. Conclusiones del capítulo . . . . .	16
<b>5. Herramientas y metodología</b>	<b>17</b>
5.1. Herramientas . . . . .	17
5.1.1. Visual Studio Code . . . . .	17
5.1.2. Trello . . . . .	18
5.1.3. Github . . . . .	18
5.1.4. Cordova . . . . .	18
5.1.5. Postman . . . . .	19
5.1.6. Android Studio . . . . .	20
5.1.7. Notepad++ . . . . .	20
5.1.8. MobaXterm . . . . .	21
5.2. Metodología . . . . .	21
5.3. Conclusiones del capítulo . . . . .	22
<b>6. Arquitectura del proyecto</b>	<b>23</b>
6.1. Cliente . . . . .	24
6.1.1. Cliente (móvil) . . . . .	24
6.1.2. Cliente (web) . . . . .	24
6.2. Servidor back-end . . . . .	24
6.2.1. Controladores y rutas . . . . .	25
6.3. Bases de datos en la arquitectura . . . . .	26
6.4. Conexión OCR . . . . .	26
6.5. Conclusiones del capítulo . . . . .	27
<b>7. Diseño</b>	<b>28</b>
7.1. Diagramas de flujo . . . . .	28
7.2. Base de datos . . . . .	36

---

7.3. Interfaz . . . . .	38
7.3.1. Edición final . . . . .	38
7.4. Conclusiones del capítulo . . . . .	59
<b>8. Tecnologías</b>	<b>60</b>
8.1. Tecnologías . . . . .	60
8.1.1. Node.js . . . . .	60
8.1.2. MongoDB . . . . .	63
8.1.3. Latex . . . . .	64
8.1.4. OpenVPN . . . . .	64
8.2. Conclusiones del capítulo . . . . .	64
<b>9. Conclusiones</b>	<b>65</b>
9.1. Conclusiones . . . . .	65
9.2. Conclusions . . . . .	66
<b>10.Trabajo futuro</b>	<b>68</b>
10.1. Trabajo futuro . . . . .	68
<b>Bibliografía</b>	<b>72</b>

# Capítulo 1

## Introducción

La piel, especialmente la de la zona de la cara, está en constante contacto con el exterior, quedando así expuesta a todo tipo de bacterias, por eso es importante atender a los cuidados y tratamientos que necesita.

Esto sumado a que España es el 5º mercado europeo que más productos de perfumería y cosmética consume con la cifra de cientocincuenta euros al año, sitúa a los consumidores españoles por encima de la media europea de ciento treinta y siete euros.<sup>1</sup>

Teniendo en cuenta que el 52 % del consumo se concentra en este tipo de productos según los datos de Stanpa la cual concluye que los hábitos de cuidar nuestra piel empiezan a ser constantes y crecientes año a año.

---

<sup>1</sup><https://www.stanpa.com/notas-prensa/el-consumo-de-perfumes-y-cosmeticos-crece-por-cuarto-ano-consecutivo-hasta-los-6-954-millones-e/>

## 1.1. Motivación

Las motivaciones principales de realizar este proyecto fue el notorio crecimiento del consumidor de productos de belleza en los últimos años y el cómo la desinformación puede dañar el cuidado de la piel.

Son muchos los *beautyvloggers/influencers* que promocionan el cuidado de la piel, el maquillaje y distintos productos cosméticos, y son aún más las personas que consumen este tipo de contenido en internet, ya sea en formato vídeo o mediante las redes sociales.

El principal problema de este tipo de recomendaciones es, desde mi punto de vista, perjudicial para personas que no conozcan del todo su tipo de piel o que sí la conozcan pero no sepan con certeza si ese producto recomendado es bueno para ellos o no.

A la hora de buscar información sobre un producto es avasallante la cantidad de contenido para discernir si es bueno o es perjudicial. Influye con bastante importancia el tema de los ingredientes ya que son abundantes y una persona de a pie no reconoce si puede o no añadirlo a su rutina de cuidado.

Los ingredientes juegan un papel importante en el desarrollo de esta aplicación, ya que serán los contrastados en el momento en el que el usuario quiera realizar una consulta para saber si el producto que quiere consumir el usuario es nocivo o no para su piel. Por tanto los ingredientes y la interacción con ellos es fundamental para el desarrollo de ésta.

Como consecuencia se me ocurrió éste proyecto, que pretende acercar al consumidor con el producto basándose en su tipo de piel.

## 1.2. Objetivos

El objetivo principal de este proyecto es aportar una solución a los problemas previamente expuestos, una aplicación útil para los consumidores de productos cosméticos para

mejorar así las condiciones del cuidado de la piel.

Seguidamente presento los objetivos para el desarrollo del proyecto:

- Estudiar la complejidad y funcionamiento de las tecnologías desarrolladas para la lectura de imágenes mediante OCR.
- Diseñar y programar una página web que permita las consultas a los productos desde el ordenador.
- Diseñar y programar una aplicación en Android que se comuniquen con la página web, pero que sea más fácil de manejar para el usuario desde el teléfono.
- Diseñar y aplicar una arquitectura que se adapte a mi proyecto.
- Definir casos de uso reales que aporten valor a la aplicación.

### 1.3. Plan de Trabajo

Para la planificación del proyecto he utilizado la aplicación de escritorio *Trello* debido a la simplicidad para realizar la organización en línea del mismo.

En la siguiente captura se puede encontrar la distribución del trabajo en cuatro grandes bloques: **Todas las tareas** (delimitadas con un color morado), **Iniciando** (con el color naranja asignado), **En proceso** (con el color amarillo) y por último **Finalizado** (con el color verde).

De esta forma me permitía saber tanto a mi como a mi tutora, cuáles eran las siguientes tareas a realizar, cuáles estaba realizando en ese momento y cuáles estaban ya finalizadas.

Cada una de las tarjetas tienen comentarios acerca del desarrollo del trabajo o anotaciones además de una fecha límite.

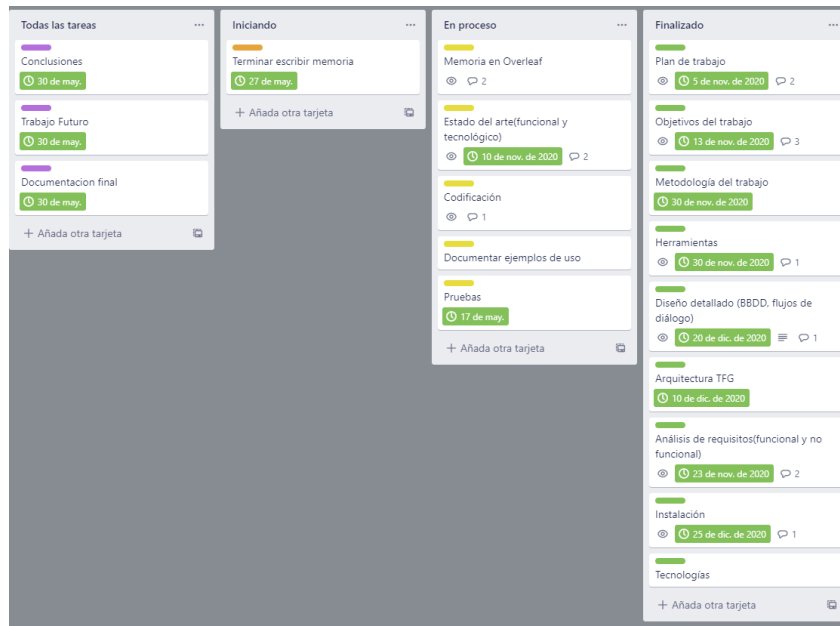


Figura 1.1: Plan de trabajo en Trello

Además de utilizar esta herramienta, me apoyaba en un documento en el cual iba anotando pasos futuros a la vez que realizaba otras tareas para poder ir marcando como hechas y finalizadas en ambos sitios, pero en este último documento que indico de una manera más desarrollada que la que se puede observar en el *Trello*.

He tenido reuniones con la tutora del trabajo cada semana o cada dos semanas en la cual hablabámos de los avances del proyecto y solventábamos dudas acerca del mismo.

## 1.4. Conclusiones del capítulo

En conclusión, la motivación de realizar el proyecto es poder aportar un servicio que desde mi punto de vista es escaso en estos tiempos y que podría ayudar a muchos usuarios.

Además, gracias a un buen plan de trabajo hablado y trabajado con la tutora de este proyecto, se ha podido realizar con el tiempo suficiente y pudiendole dedicar toda la atención que ha requerido. Eso junto con los objetivos claros desde un primer momento, considero que han dado lugar a una aplicación funcional.

# Capítulo 2

## Introduction

The skin, especially that of the face area, is in constant contact with the exterior, which means that collect all the bacteria, that is why it's important to attend to the care and treatments it needs.

This, added to the fact that Spain is the 5th European market that consumes the most perfumery and cosmetic products and cosmetics with a quantity of one hundred and fifty euros per year, places Spanish consumers above the European average of one hundred and thirty seven euros.

Taking into account that 52 % of the consumption is concentrated in this type of product according to Stanpa's data, it concludes that the habits of caring for our skin are beginning to be constant and increasing year after year.

### 2.1. Motivation

The main motivation for carrying out his project was the notable growth of the consume of beauty products in recent years and how misinformation can damage skin care.

There are many *beautyvloggers* / *influencers* who promote skin care, makeup and different cosmetic products, and there are even more people who consume this type of content on

the Internet, either in video format or through social networks.

The main problem with this type of recommendation is from my point of view, harmful for people who don't fully know their skin type or who do know it but don't know for sure if that recommendation of the product is good for them or not.

When looking for information about a product, the amount of content is overwhelming to discern if it is good or harmful. The issue of ingredients influences quite importantly since they are abundant and an ordinary person does not recognize whether or not they can add it to their care routine.

As a consequence, I came up with this project, which aims to bring the consumer closer to the product based on their skin type.

## **2.2. Objectives**

The main objective of this project is to provide a solution to the previously exposed problems, a useful application for consumers of cosmetic products to improve skin care conditions.

Next I present the objectives for the development of the project:

- Study the complexity and operation of the technologies developed for image reading with OCR.
- Design and program a web page that allows inquiries to the products from the computer.
- Design and program an Android application that communicates with the website, but is easier for the user to use from the phone.
- Design and apply an architecture that suits my project.
- Define real use cases that add value to the application.



## 2.3. Work-plan

For the planning of the project I have used the desktop application *Trello* due to the simplicity of organizing it online.

In the following screenshot you can find the distribution of work in four large blocks: **All tasks** (delimited with a purple color), **Starting** (with the assigned orange color), **In process** (with yellow color) and finally **Finished** (with the green color).

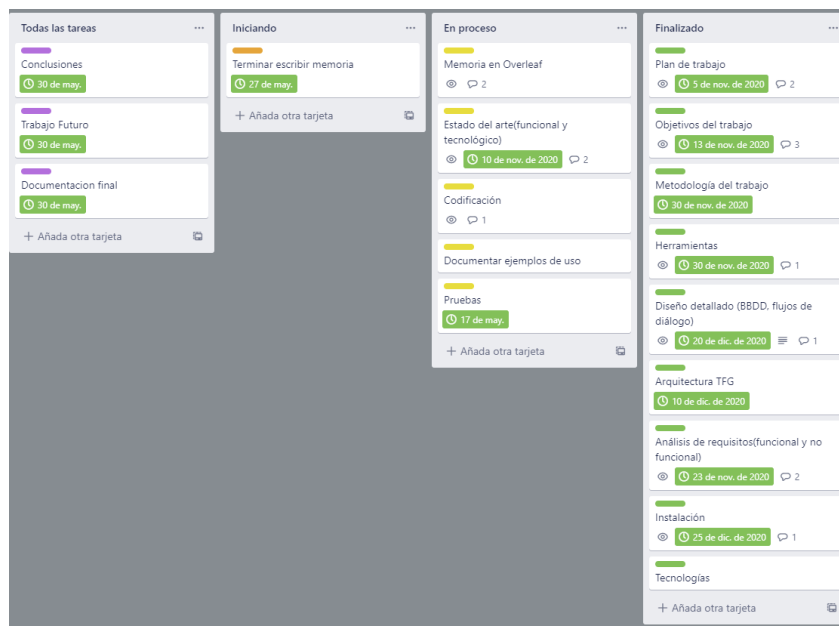


Figura 2.1: Working plan on Trello

In this way it allowed me to know both myself and my tutor, what were the next tasks to be done, which ones I was doing at that time and which ones were already finished.

Each of the cards has comments about the progress of the work or notes as well as a deadline.

## 2.4. Chapter conclusion

In conclusion, the motivation of perform this project is to contribute a service that by my point of view is scarce in our times, and that could help a lot to many users.

In addition, thanks to a great work-plan that has been discussed and worked with the tutor of this final project, it has been possible to make this with the necessary amount of time and being able to dedicate full attention that it requires. That together with a clear objectives from the beginning, I consider that they will lead to a functional application.

# Capítulo 3

## Estado del arte

En esta sección de la memoria expongo un estudio de las aplicaciones o servicios que tengan similitudes con el trabajo que he realizado.

Así mismo voy a exponer las diferentes tecnologías que me han ayudado a cumplir dicho objetivo además de exponer las que he descartado y el porqué del mismo.

Por último haré una conclusión con las tecnologías finales con las que he desarrollado el proyecto.

### 3.1. Aplicaciones similares

#### 3.1.1. Clean Beauty

Esta aplicación francesa, creada por un equipo de científicos, permite identificar los ingredientes dañinos, alérgenos o prohibidos en Europa.

Los productos que analiza son fundamentalmente de higiene y belleza, lo que incluye maquillaje, esmalte de uñas o tintes para el pelo.

La metodología de identificación es mediante una fotografía a la lista de los ingredientes añadida en los envases de los productos a analizar.

Analizando las opiniones respecto a la aplicación podemos observar que el principal fallo de dicha aplicación es que no puedes importar imágenes previamente hechas, tienen que

ser al momento, lo que dificulta por ejemplo las comprobaciones de los productos previamente comprados.

Por otro lado, no anuncia que ha leído completamente la lista de los ingredientes, lo que deja al usuario con la duda.

Los puntos positivos que valoran más los usuarios es que la aplicación está en Español, lo que facilita el entendimiento de ésta. Además, los análisis y comprobaciones están basados y corroborados por datos y citas científicas lo que añade confianza a la hora de confiar en las valoraciones aportadas sobre los ingredientes dañinos o prohibidos.

### **3.1.2. 1Source**

1Source es una de las aplicaciones más descargadas a nivel internacional, al igual que la anterior aplicación revisa los ingredientes de los cosméticos aunque también analiza otro tipo de productos alimenticios, de limpieza, para mascotas entre otros.

Utiliza un sistema de medidas del cero al diez agrupados por colores y se basan en las prohibiciones de la Unión Europea para aplicar el máximo puntaje y el color rojo a los productos prohibidos por dicha regulación.

La versatilidad de esta aplicación en concreto reside en que la subida de fotografías puede ser instantánea, subida de la galería del usuario ya sea una imagen o una captura de pantalla además de códigos de barras.

Alerta la utilización de los productos escaneados en referencia a si el consumidor tiene problemas previos cutáneos.

A diferencia de la aplicación previa ésta sí enumera los diferentes ingredientes que ha analizado para que el usuario se cerciore de que están todos leídos.

El principal problema de ésta solución es que está íntegramente en Inglés lo que causa dificultad a muchos hispanohablantes, sumado a que la aplicación está desarrollada en China y por tanto tiene algunos productos en su idioma.

La cantidad de funcionalidades de ésta aplicación la hace excesiva desde el punto de vista del usuario y por tanto liosa a la hora de utilizarla.

## 3.2. Aplicaciones web similares

### 3.2.1. Inci

Inci es una plataforma que ayuda al cliente con los ingredientes perjudiciales de los productos fundamentalmente cosméticos.

Es un buscador de los ingredientes INCI en español, desarrollado por 'nuilea.com'.

La página web cuenta con diferentes secciones de los productos analizados, además de los cuidados de la piel del rostro añade categorías como maquillaje, productos para bebés o cremas solares. De éstos productos analizados realiza 'top's' de algunas de las categorías.

El punto fuerte de la aplicación es sin duda que está dedicado para la comunidad hispanohablante, lo que hace más fácil la lectura y entendimiento de la información buscada acerca de los productos de interés del consumidor.

No se pueden analizar productos en concreto mediante fotografías, sólo mediante el buscador que tiene la página. En el cual puedes realizar la búsqueda de ingredientes concretos o marcas.

La manera de analizar los ingredientes es mostrando el nombre y añadiendo al lado un color para mostrar si es o no perjudicial.

Por tales motivos, la aplicación web está orientada de una manera muy general ya que no lleva un registro del usuario, ni de los productos utilizados por el mismo ni de las complicaciones o características del mismo para poder recomendar o no los productos que busca.

### 3.3. Tecnologías de OCR

En esta sección voy a plasmar las diferentes aplicaciones, APIs o librerías que he encontrado acerca de la tecnología que he utilizado para realizar mi trabajo.

#### 3.3.1. ¿Qué es el OCR?

El OCR es un sistema computarizado de análisis que permite escanear un documento de texto en un fichero automatizado.

La traducción en español es **Reconocimiento Óptico de Caracteres (ROC)**, y tiene como fin el reconocimiento de caracteres de texto impreso o la digitalización del mismo.

Estos sistemas de reconocimiento tienen la capacidad de reconocer tipos de fuentes de texto así como caracteres de imprenta de máquinas de escribir u ordenadores.

Algunos de los sistemas OCR más avanzados son capaces incluso de reconocer e identificar la escritura manual.

La principal razón de apoyarme en este sistema para mi aplicación, es que me permite realizar una lectura de una fotografía, extrayendo así de ésta un texto plano de los ingredientes que componen el producto que el usuario desea comprobar.

Una vez hayado el texto plano y la lista de los ingredientes, la aplicación obtiene un resultado en función de si esos ingredientes son beneficiosos o no según el tipo de piel que haya especificado dicho usuario.

#### 3.3.2. Tecnologías de OCR descartadas

**Google Vision:** Una de las aplicaciones o bibliotecas consideradas en un primer momento. Debido a los siguiente problemas he decidido no incluirlas en mi proyecto.

En un primer lugar al buscar información aparecía la conocida *Google Vision*. Otra biblioteca más reciente para el escaneo de los mismos códigos previamente citados en la primera aplicación, pero además de eso da soporte a otro tipo de servicios

como la detección de rostros o reconocimiento de texto.

El principal problema es que esta versión está anticuada y se va a acabar el soporte a ésta de un momento a otro ya que no han anunciado una fecha específica.

Y todo esto se debe a que han desarrollado un nuevo kit para el aprendizaje de *machine learning*. Éste kit cuenta, entre otras cosas, con esta herramienta de *Google Vision* pero con un nombre diferente y la cuál, según Google siguen desarrollando.

Por lo que no se hacen responsables ni del soporte a la antigua versión, ni de los problemas que puedan dar los parches con actualizaciones a su nueva versión.

**Computer Vision Read Azure:** En este caso, esta API desarrollada por Azure es una de las más potentes y exitosas del momento y por tanto la más competitiva.

Permite extraer el texto de muchas más fuentes por ejemplo fotos de matrículas, contenedores con números de serie o diferentes documentos. Además, soporta la lectura de varios documentos extensos con diferentes lenguajes, aunque el texto manuscrito solo puede realizar la lectura si es en inglés.

Tiene una amplia lista de formatos que soporta: PEG, PNG, BMP, PDF y TIFF. Al igual que en la página web anterior, cuenta con mucha documentación lo que haría más fácil su uso.

El inconveniente para no utilizar en concreto esta de Azure es que no soporta Node.js.

### 3.3.3. Tecnología OCR utilizada

#### OCRSpace

Esta API, ofrecida por OCRSpace, permite una manera fácil y sencilla de analizar las imágenes, adecuado para la finalidad de mi aplicación.

El resultado de leer dichas imágenes se obtiene en un formato JSON con el texto extraído de éstas. Dicha API tiene una ejecución en tres servidores físicamente distanciados y tiene la opción en versión web o descargable.

Las ventajas de esta API de esta web es que hay ejemplos de codificación, muchos planes gratuitos y además explicaciones de todo el funcionamiento.

Además soporta una de las tecnologías en las que quiero desarrollar el proyecto: *node.js*.

### 3.4. Bases de Datos

En cuanto a las bases de datos, he decidido alojar todo mi contenido en una estructura de MongoDB. Hay diferentes opciones y múltiples gratuitas. Entre ellas:

**MongoDB Atlas:** Finalmente para administrar todo el proceso de la base de datos me he decantado por MongoDB Atlas porque es un servicio en línea y no requiere de ninguna instalación como si lo requería una de las herramientas descartadas. MongoDB Atlas es el primer servicio DaS, *Database as a Services*, para bases de datos de mongo, que es justo la que he seleccionado yo.

Gracias a esta herramienta, se pueden crear diferente clústers de bases de datos y exponerlos en internet para conectar con las aplicaciones. El poder que considero que tiene esta herramienta es que la gestión y administración de los datos así como de la propia base de datos es directamente desde la página web por lo que la plataforma es más accesible.

Además, es fácil e intuitiva, y para consultas externas que no se pudiesen realizar directamente en la web, he utilizado una conexión mediante línea de comandos conectada a un clúster.

El diseño de la misma así como el contenido se explica en los puntos de más adelante en la memoria.



### 3.5. Conclusiones del capítulo

Gracias al estado del arte he podido obtener ideas e información de otras aplicaciones que pueden o podrían dar el mismo servicio que ofrece mi aplicación.

Dandome cuenta así de las faltas de algunas de ellas, que he podido mejorar en mi aplicación para poder hacer así una distinción entre otros servicios ya existentes y el mío.

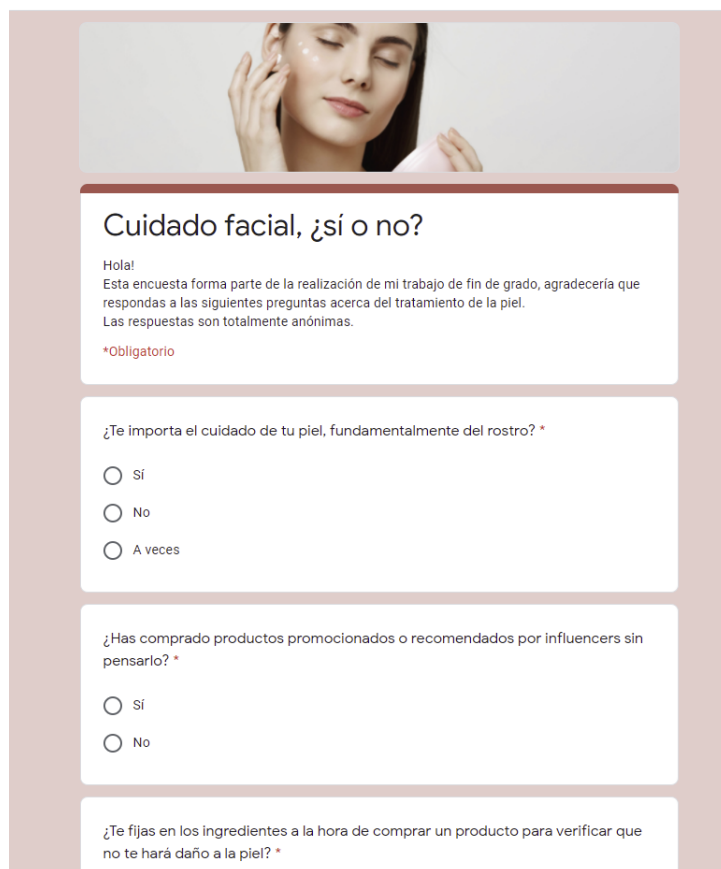
A la hora de investigar acerca de las tecnologías que podía utilizar para mi proyecto, me he topado con algunas de ellas muy interesantes, como ya comento, pero que para los requisitos de mi aplicación se quedan o bien escasos o bien no les dan soporte en estos momentos, lo que para mí suponía una falta de seguridad para la aplicación.


Basándome en esos criterios he decidido cuales serían las que mejor se adaptaban a la hora de realizar las funcionalidades del proyecto.

# Capítulo 4

## Escenarios y requisitos

Con la ayuda de los formularios realizados a distintos usuarios he sacado información clave para el diseño y realización de los requisitos.





### Cuidado facial, ¿sí o no?

Hola!  
Esta encuesta forma parte de la realización de mi trabajo de fin de grado, agradecería que respondas a las siguientes preguntas acerca del tratamiento de la piel.  
Las respuestas son totalmente anónimas.

**\*Obligatorio**

¿Te importa el cuidado de tu piel, fundamentalmente del rostro? \*

☐ Sí

☐ No

☐ A veces

¿Has comprado productos promocionados o recomendados por influencers sin pensarlo? \*

☐ Sí

☐ No

¿Te fijas en los ingredientes a la hora de comprar un producto para verificar que no te hará daño a la piel? \*

Figura 4.1: Foto encuesta realizada

Para esclarecer la utilidad de la aplicación hablaremos acerca de los distintos escenarios en los que el usuario utilizaría *HiBeauty!*.

Además se explicarán los requisitos funcionales con los que va ha contado este proyecto.

## 4.1. Escenarios

El objeto de esta sección es plasmar escenarios reales en los que el usuario necesite de las funciones de la aplicación desarrollada para este proyecto.

**Primer escenario. Escaneando producto en tienda por recomendación.** Visualizando las redes sociales el usuario se topa con un influencer que recomienda un producto cosmético, pero no indica si se puede utilizar para determinado tipo de piel y tampoco facilita ningún tipo de información acerca de los ingredientes que componen el mismo. El usuario está interesado en el producto pero necesita saber más acerca del producto para decidir si realiza la compra o no.

El usuario realiza una captura del producto que muestra el personaje público. Cuando se acerca a la tienda para comprar el producto, gracias a la tecnología de la aplicación puede escanear los ingredientes y al tener registrado su tipo de piel, ésta muestra en un rango cuánto de dañino es para su piel dicho producto, ayudando así al usuario a decidir sobre la compra o no del producto.

**Segundo escenario. Escaneando productos previamente comprados.** En esta ocasión el usuario ya tiene varios productos comprados, pero es principiante en el mundo "beauty" para el cuidado del rostro y de los cosméticos faciales, por lo que decide consultar la aplicación.

Uno a uno escanea los productos haciéndole foto instantánea para poder decidir en función de su registro previo del tipo de piel, si le vienen bien o no y si los puede seguir usando en su rutina.

**Tercer escenario. Añadiendo tipo de piel a la aplicación.** Nuevamente el usuario ha visto productos que le interesan en las RRSS, por lo que le gustaría conocer si éstos dañan o no su piel, fundamentalmente la del rostro. Para ello primero descarga la aplicación y selecciona el tipo de piel que tiene en las distintas zonas de la cara para así poder comenzar con el escaneo y utilización de los productos.

## 4.2. Requisitos funcionales

Apoyándome en los escenarios anteriores he analizado y decidido cuáles han sido las funcionalidades finales que la aplicación va a proporcionar al usuario.

Los servicios van a estar seccionados en las distintas facciones que tiene el proyecto, en cada una de ellas se va a describir a modo general y al mismo tiempo se van a explicar las funciones concretas de cada una.

### 4.2.1. Productos

En el apartado de los productos se pueden identificar tres funcionalidades distintas, que explicaré detalladamente en cada uno de sus puntos.

Este apartado hace referencia a los productos que el usuario ha escaneado previamente, ya sea para consultar la totalidad de los mismos indistintamente del resultado del análisis de la aplicación así como los productos recomendados gracias a dicho análisis. Además el usuario tendrá la opción de añadir productos a una lista de favoritos.

#### **Productos consultados**

Además de los productos recomendados se hace un registro de todos los productos consultados por el mismo, por lo que en esta nueva lista se exponen los nombres de éstos además junto con la lista de ingredientes del mismo.

El producto debe de ser guardado previamente por el usuario.

### **Lista de favoritos**

Aquí el usuario podrá añadir los productos que el usuario considere que quiere guardar en esta categoría.

### **4.2.2. Cámara**

En esta sección de cámara hablaremos sobre cómo tomar una fotografía instantánea para su análisis.

#### **Instantánea**

Si la foto la realiza directamente desde el teléfono, tendrá la opción de repetirla si ha salido movida, ya que cuanto menos movida esté la fotografía más fácil será obtener los ingredientes de la lista. A continuación, se mostrará una lista de los ingredientes nocivos para el usuario.

Una vez resuelto el análisis por parte del algoritmo que decide si los ingredientes son perjudiciales, se mostrará por pantalla un porcentaje y un color en función de cuán de beneficioso es para el usuario la utilización de este producto.

Una vez se ha mostrado la información, el usuario tendrá la opción de introducir el nombre del producto y de añadirlo a la lista de productos guardados y a su vez a una lista de favoritos.

### **4.2.3. Usuario**

Por último, en el apartado del usuario, se divide en tres funcionalidades distintas, que serán detalladas en sus respectivos puntos. Consiste en la configuración de todo tipo del perfil del usuario, ya sea en cuánto al tipo de piel o en cuanto a la configuración de su perfil, para tener una mayor compactibilidad.

### Seleccionar tipo de piel

Para facilitar el uso de la aplicación y que sea entendible, al seleccionar esta opción en el perfil del usuario, aparecerán tres zonas de la cara con tres opciones. Las zonas son: *zona T*, *zona mejillas* y *zona barbilla*. Y las pieles a elegir, de las cuales he tratado de elegir las más generales: grasa, seca y/o sensibles.

Una vez seleccionado el tipo de piel, el usuario debería pulsar el botón de *Enviar*, tras el cual aparecerá una alerta para informar al usuario de que la piel ha sido añadida.

Se necesitará tener todas las zonas completadas antes de realizar un análisis.

### Configuración perfil

Para que el usuario pueda configurar su perfil y administrar su cuenta de la manera más clara posible. Entre estas opciones, el usuario podrá cambiar su contraseña, su nombre y su correo electrónico si así lo desea.

Además podrá informar de algún problema de la aplicación o el servicio. Y por último, el usuario podrá cerrar la sesión con su cuenta actualmente activa y en otra de las opciones, eliminar la cuenta de una manera lógica, ya que el usuario no puede acceder a la base de datos.

## 4.3. Conclusiones del capítulo

Para finalizar y desde mi punto de vista, he seleccionado los escenarios que más se han podido acercar a la realidad del uso de la aplicación y he intentado reflejarlos de la manera más clara y concisa posible, pensando en toda las posibilidades que ofrecía la aplicación.

Lo cual me ha ayudado a decidir de una manera más transparente las funcionalidades que iba a tener mi aplicación y el camino por la cual quería llevar la misma. Así pues obtuve los requisitos que debía de tener el proyecto y así los he reflejado en las secciones anteriores.

# Capítulo 5

## Herramientas y metodología

### 5.1. Herramientas

En esta sección voy a exponer las diferentes herramientas en las cuales me he apoyado para realizar el proyecto.

#### 5.1.1. Visual Studio Code

Esta herramienta es un editor de código fuente soportado por múltiples sistemas operativos como Windows, Linux y macOS. Desarrollado por Microsoft, incluye la funcionalidad de depurar el código.

Además contiene el Git integrado, finaliza inteligentemente el código y realiza el resaltado del código, entre otras muchas ventajas de este editor.

Además de todas esas opciones, el usuario puede instalar extensiones concretas para diferentes lenguajes o funciones específicas para un proyecto.

Cabe destacar que es totalmente gratuito y de código abierto.

Los fines con los que he usado esta herramienta son dos, en un primer lugar arrancaba el servidor de *node.js* y en el segundo y principal lugar, es donde he desarrollado el código.

### 5.1.2. Trello

Esta aplicación sirve para la gestión de proyectos más utilizada actualmente.

Usualmente la aplicación es compartida por un grupo para la organización y distribución de tareas, en mi caso se ha compartido con la tutora del proyecto para un seguimiento del desarrollo así como de las fechas asignadas para las tareas.

### 5.1.3. Github

Github es una de las principales plataformas creadas para el seguimiento y desarrollo de proyectos abiertos de herramientas o aplicaciones.

Se caracteriza por las funciones colaborativas que ayudan a todos los implicados en un proyecto a llevar un registro del avance del proyecto.

Esta herramienta se ha utilizado para la subida del código del proyecto, ya que así se puede observar el avance del proyecto con los commits realizados.

### 5.1.4. Cordova

Apache Cordova es un framework de código abierto utilizado para el desarrollo de aplicaciones móviles fundamentalmente.

El principal punto de este entorno es que es multiplataforma, lo que quiere decir que con el mismo código puedes utilizarlo para distintas plataformas móviles como Android, windows phone e incluso para la página web.

La mayoría de los proyectos desarrollados en esta plataforma han sido utilizando JavaScript, HTML5, CSS3. Esta API nos permite acceder mediante JavaScript a características nativas del móvil como la cámara que en este caso es lo que nos interesa, pero se puede acceder a más de éstas como los contactos, los eventos o la geolocalización.

Para la plataforma móvil utilizada, se incluye un núcleo Web como parte de la aplicación nativa, de esta forma se genera una aplicación "nativa" para cada plataforma elegida por



el desarrollados, aprovechando para todas el mismo núcleo de la aplicación.

Las aplicaciones desarrolladas con Apache Cordova se consideran aplicaciones híbridas, es decir, que es una aplicación nativa con una capa intermedia de herramientas que hacen uso de otros lenguajes de programación.

La finalidad de utilizar esta herramienta es poder realizar un sólo código central para la aplicación móvil y la aplicación web de este proyecto, y así no desarrollar doblemente en distintos lenguajes.

### 5.1.5. Postman

Esta plataforma se utiliza para facilitar los pasos a la hora de crear una API.

En ella se puede elegir el servicio que mejor se adapte a las necesidades del proyecto.

Entre ellos están:

- Crear, enviar y guardar solicitudes de tipo REST, SOAP o GraphQL
- Editar URLs
- Editar encabezados de solicitudes
- Guardar encabezados preestablecidos
- Gestionar cookies

A la hora de integrar Postman para la realización de mi aplicación, ha sido para poder enviar peticiones POST y GET al servidor de una manera más rápida y eficiente. Permitiéndome así el poder gestionar problemas o el desarrollo de funcionalidades que necesitasen una petición previa para su correcto funcionamiento y evitar así el largo proceso de lanzar la aplicación cada instante que necesitase realizar una comprobación.

Además, en esta plataforma se pueden recoger las respuestas que realizaban las peticiones, ya sea en forma de json o en forma html. Pudiendo así visualizar de una manera más rápida y concreta el estado de mi aplicación.

### 5.1.6. Android Studio

Esta herramienta es un entorno de desarrollo integrado (IDE) oficial para el desarrollo de apps de Android. Es un potente editor de códigos con unas características exclusivas que ayudan al desarrollador en su labor.

Entre ellas:

- Un sistema de compilación flexible basado en Gradle
- Un emulador rápido y con diferentes funciones
- Un entorno unificado para el desarrollo de varios dispositivos Android
- Integración con Github

Android Studio ha sido un factor clave en el desarrollo del proyecto. Me he apoyado en él para la utilización de los emuladores móviles. Necesitaba un gestor de AVD, en el cual poder emular mi aplicación como si de un dispositivo móvil se tratase para comprobar el funcionamiento de la misma.

El emulador cuenta con todo tipo de opciones que he usado en beneficio, como el emulador de la cámara, la galería de fotos entre otras.

### 5.1.7. Notepad++

A parte del ya mencionado Visual Studio Code, me he apoyado en este otro editor de código gratuito para poder llevar en paralelo algunas acciones. Cabe destacar que este editor cuenta con pluggins para por ejemplo comparar una serie de archivos entre ellos y resaltar las diferencias.

Es fácil, rápido y realmente me ha ayudado a distinguir las tareas, porque el Visual Studio Code lo utilizaba única y exclusivamente para el desarrollo del código.

### 5.1.8. MobaXterm

MobaXterm es una herramienta de Windows para el manejo del servidor. Permite la conexión remota con inicio de sesión de usuario, además de conexiones FTPs o SSH, utilizado con comandos Unix.

A la hora de ayudar al project, MobaXterm es la conexión con el servidor y es la más intuitiva y visualmente fácil que he podido encontrar. También cabe destacar que la posibilidad de una conexión STFP ha ayudado a la subida de los archivos al servidor de una manera cómoda y sencilla.

## 5.2. Metodología

Tras contemplar todos los tipos de metodologías para la gestión de proyectos, en este caso de carácter tecnológico, he decidido que la mejor solución para mi proyecto es utilizar la metodología híbrida.

Ésta consiste en una implementación de las dos más conocidas: conservadora/waterfall y la metodología ágil.

Las principales características por las cuales me decanté por esta metodología son:

- Fácil adaptación

Con este método, la adaptación a las necesidades del proyecto tienen que ser fáciles, si surge cualquier problema con el desarrollo en mitad del proyecto, necesito tener una adaptación fácil al error.

- Planificación

Esta característica es la base para la organización del proyecto así como de los recursos necesarios para el desarrollo del mismo.

Además ayuda a coordinar las distintas actividades y estar controlando como avanza el mismo.

Es por eso que considero que una buena planificación del proyecto de una manera más

conservadora puede aportar una base sólida a la vez que necesito poder retroceder a algún paso en el que haya un fallo de una manera rápida y fácil, y por ambas cosas he decidido que la metodología que más se adapta a mis necesidades es la metodología híbrida.

### **5.3. Conclusiones del capítulo**

En el momento de escoger las herramientas para el desarrollo de la aplicación, he seleccionado las que más me han interesado aprender así como las que ya he utilizado a lo largo de la carrera. De este modo he podido reflejar mi conocimiento sobre algunas de ellas, ya que ya sabía utilizarlas y las he querido sacar provecho ya que me valía perfectamente para el desempeño del trabajo.

Así mismo he aprendido a utilizar algunas nuevas herramientas necesarias para el proyecto ya que aportaban novedad y sobretodo agilidad, además de poder nutrirme personal y profesionalmente.

Al igual que para escoger las herramientas, la metodología ha sido seleccionada en función del tiempo y el proyecto que yo iba a desarrollar.

## Capítulo 6

### Arquitectura del proyecto

A continuación se van a enumerar los componentes del proyecto.

La arquitectura se ha diseñado con un conjunto de 5 módulos: cliente(desde distintos dispositivos), un servidor con el back-end, la base de datos y por ultimo la conexión del back-end con el servicio de OCR.

Se adjunta figura explicativa:

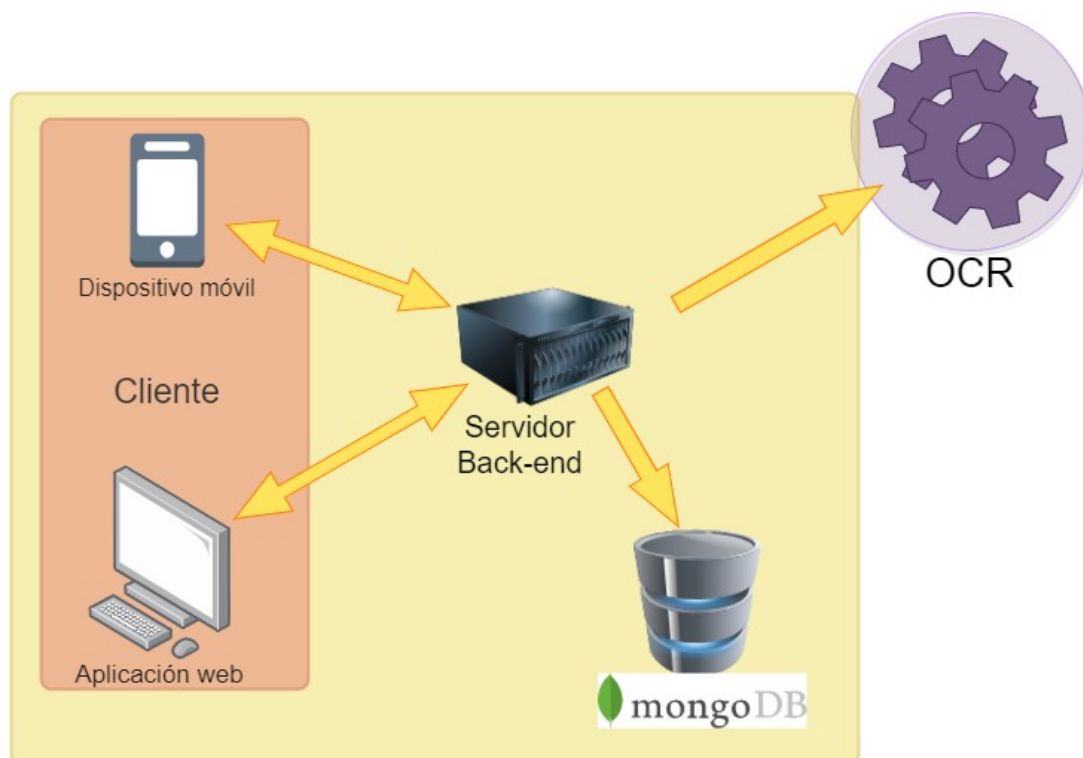


Figura 6.1: Diagrama con la arquitectura del proyecto

## 6.1. Cliente

Desde la parte del cliente común entre las dos secciones que distingo más adelante, se puede acceder a las distintas funcionalidades desde la página principal de la aplicación.

El cliente se encargará de realizar las peticiones en función de la acción que quiera realizar y será el servidor el que sirva las respuestas a éste.

### 6.1.1. Cliente (móvil)

Uno de los puntos de acceso del cliente va a ser mediante un dispositivo móvil.

El cliente envía una petición de tipo GET o POST al servidor back-end mediante una fotografía y el servidor le devuelve la respuesta por pantalla con la resolución de dicha petición.

Esta petición desde el servidor conecta con el servicio proporcionado por *OCRFree space* el cual analiza y obtiene los ingredientes de la foto que el usuario haya querido mandar a analizar.

### 6.1.2. Cliente (web)

Otro de los puntos del acceso del cliente es mediante una aplicación web conectada al igual que el anterior cliente, al servidor back-end.

Las peticiones al servidor seguirán siendo de tipo REST, pero con otro tipo de contenido ya que desde el ordenador no se podrán hacer consultas mediante fotografía instantánea.

## 6.2. Servidor back-end

En este servidor se va a desarrollar el back-end del proyecto, se encarga de traducir los datos que envía el cliente, distinguir entre las dos posibles peticiones del mismo. El servidor utiliza la tecnología de *node.js* que describo en los siguientes capítulos. Por tanto a la hora de levantar el servidor se ha utilizado la consola de windows para levantarlo con

*node.js*.

Así mismo conecta con la aplicación de servicio del OCR, analiza y descifra el contenido encapsulado en formato JSON para preparar una respuesta hacia el cliente.

Además cabe destacar que el servidor realiza también las consultas a la base de datos, conectando directamente con mongo, ya que realiza dicha conexión a la hora de arrancar el proyecto y de esta manera con los datos en la nube se facilita mucho más el obtener los datos.

Por último, el servidor es quién recibe todas las peticiones GET y POST redirigidas por las rutas que describo en la siguiente sección. En la parte del servidor se ha desarrollado todo el conjunto de funcionalidades que forman parte de la aplicación, ya sea la recuperación de datos, así como las recomendaciones de productos para el usuario.

### 6.2.1. Controladores y rutas

Los controladores se encargan de redirigir las peticiones HTTP que se realizan al servidor.

Usuario		
Logearse en la aplicación	POST	/users/singin
Registrarse en la aplicación	POST	/users/signup
Cerrar sesión	POST	/users/logout
Mostrar el nombre del usuario en la sesión iniciada	GET	/user/nombre
Eliminado lógico del usuario	POST	/user/delete
Cambiar nombre del usuario	POST	/user/cambiar_nombre
Cambiar el correo electrónico del usuario	POST	/user/cambiar_email
Cambiar contraseña del usuario	POST	/user/cambiar_password
Establecer zonas faciales	POST	/update/zonas

Productos Escaneados		
Obtener listas de productos guardados	POST	/user/lista_productos
Obtener listas de productos favoritos	POST	/user/lista_favoritos
Crear y guardar el producto deseado por el usuario	POST	/user/guardarProducto
Marcar como favorito el producto que desee el usuario	POST	/user/favorito_producto

OCR		
Realizar la llamada a OCR	POST	/OCR/read

### 6.3. Bases de datos en la arquitectura

La base de datos se conectará directamente con el servidor back-end para la realización de las queries necesarias a la hora de comprobar la información de la petición que llega desde el cliente.

El diseño y componentes de la base de datos utilizada se especifica en el siguiente capítulo.

### 6.4. Conexión OCR

Un vez el cliente tome una fotografía de la etiqueta a consultar, y dicha foto sea transmitida al servidor, éste va a realizar la consulta al servicio de OCR encargado de devolver en un formato JSON todas las palabras que aparecen en la lista de los ingredientes para su posterior análisis en el servidor.

Este servicio con el que conecta el servidor para mandarle las fotografías, cuenta con muchas posibilidades que me han ayudado considerablemente. Tiene opciones de rotar las imágenes que vienen desde el servidor, puede distinguir entre archivos corruptos y no corruptos, se puede fijar el tipo de salida que necesito e incluso la entrada, que en mi caso es en base64.



## 6.5. Conclusiones del capítulo

Finalmente, la arquitectura del proyecto es fundamental para el desarrollo del mismo, por ello he optado por una versión asequible pero funcional.

Los controladores me han ayudado a visualizar de una manera clara las peticiones que el usuario podría realizar a la hora de utilizar la aplicación.

# Capítulo 7

## Diseño

### 7.1. Diagramas de flujo

Para la implementación de la funcionalidad he realizado unos diagramas de flujo que representan algunas de las funcionalidades de la aplicación.

En esta primera figura 7.1 encontramos el diagrama de flujo consistente en el inicio de sesión del usuario, en el cual, se comprueba si el usuario está o no registrado y además, si está o no activo (flag que indica si ha sido borrado o no lógicamente).



Figura 7.1: Diagrama de inicio de sesión

En este otro flujo, se encuentra el diagrama que representa la acción del usuario al querer escanear un producto, mediante una foto instantánea desde el teléfono.

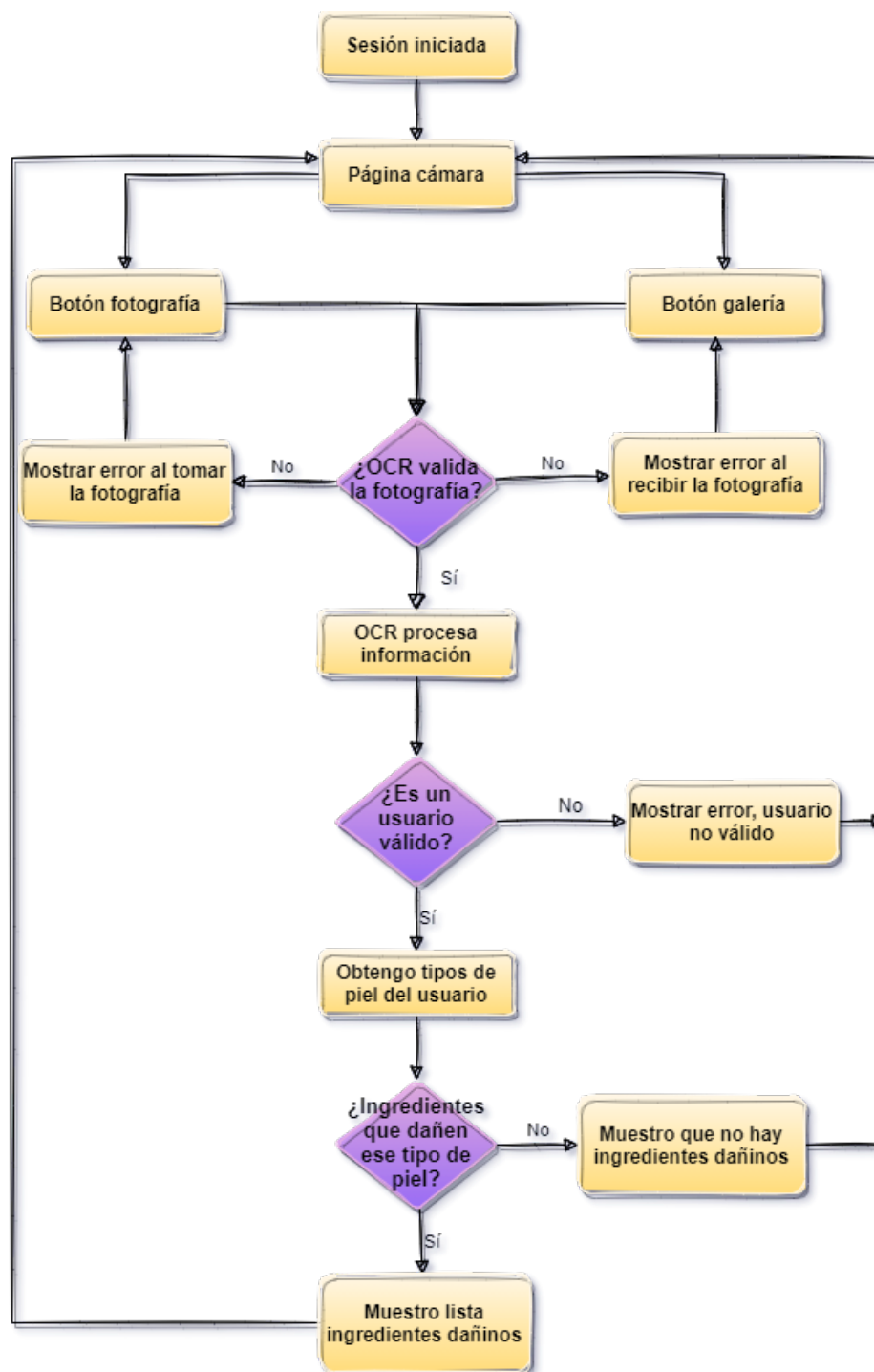


Figura 7.2: Diagrama de escaneo de una fotografía

Además, este otro diagrama plasma el flujo de la aplicación mediante el cual el usuario decide y selecciona cuáles son las preferencias para sus zonas de piel faciales.

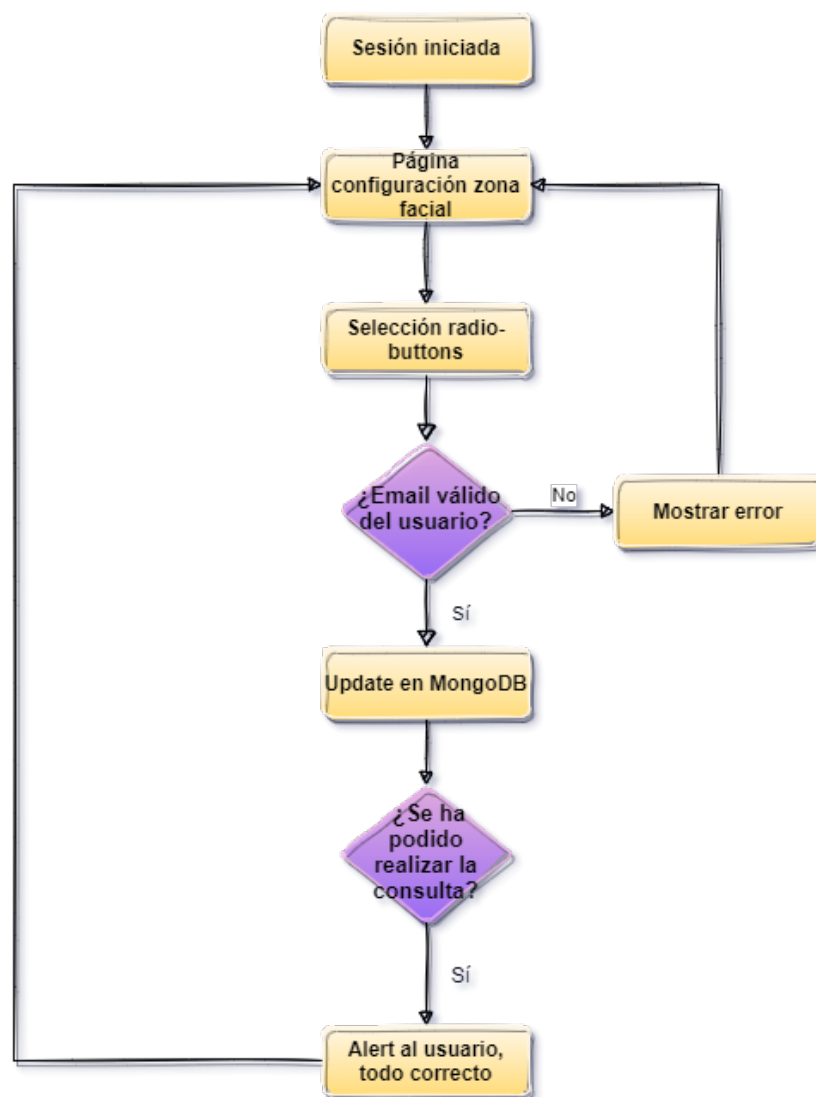


Figura 7.3: Diagrama de selección zonas faciales

Por otro lado tenemos el flujo que explica todos los pasos que sigue la aplicación para guardar un producto previamente escaneado por el usuario.

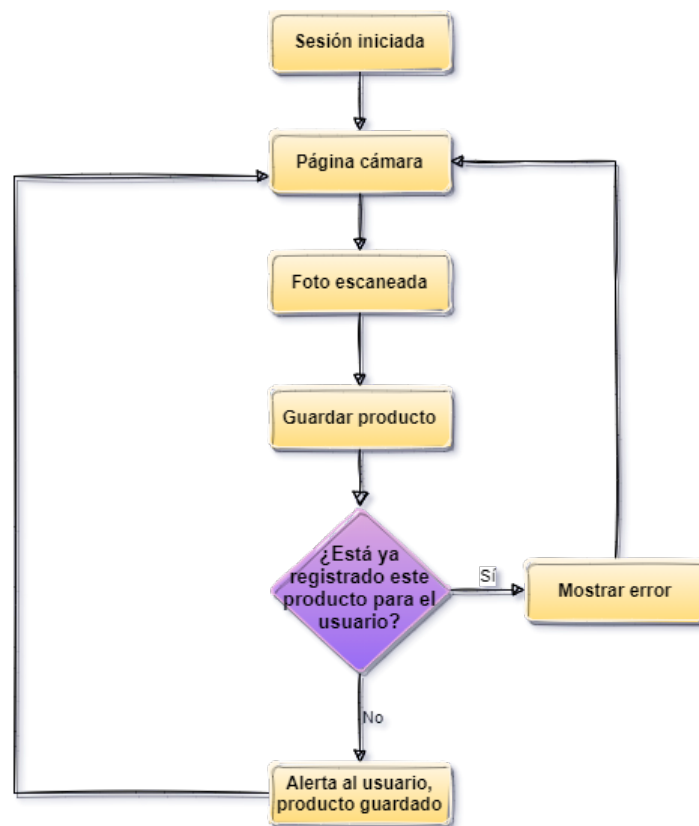


Figura 7.4: Diagrama de cómo guardar un producto previamente escaneado

En la siguiente figura 7.5, se observa el flujo que sigue la aplicación para mostrar la lista de favoritos del usuario, para ello el usuario tiene que estar previamente con la sesión iniciada y además, tiene que haber marcado como favorito alguno de los productos que ha escaneado previamente.

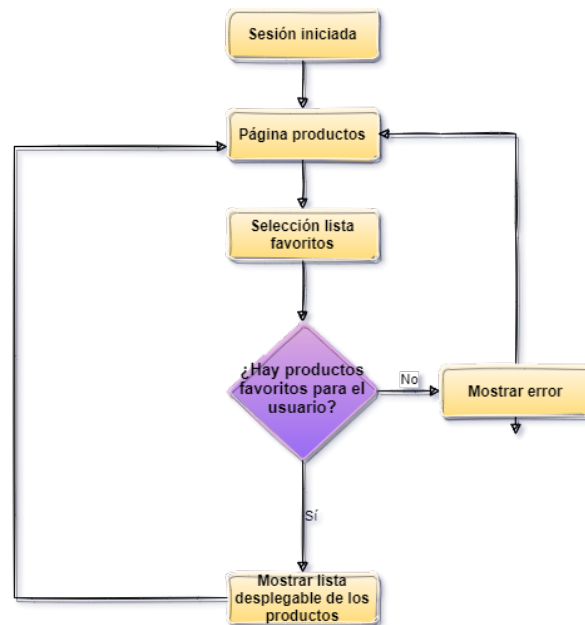


Figura 7.5: Diagrama de cómo guardar un producto previamente escaneado

El diagrama de flujo siguiente, se muestra el proceso del borrado lógico por parte del usuario, en el cual se pone a *false* el campo correspondiente del usuario que indica si la cuenta está activa o no mediante un *booleano*.

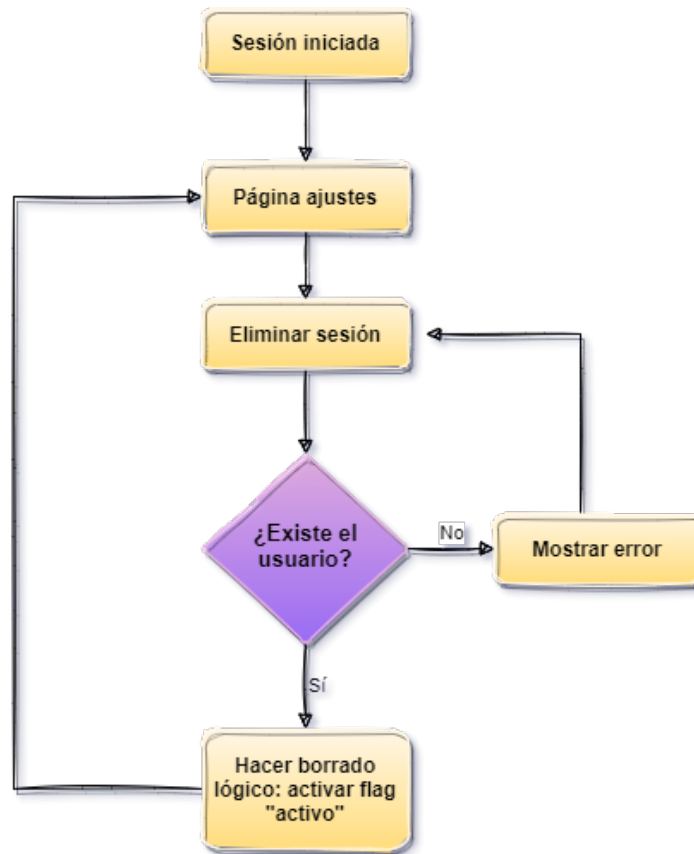


Figura 7.6: Diagrama de borrado lógico de la cuenta del usuario



En este último diagrama, se indica el flujo del cambio de una contraseña por parte del usuario, la cual tiene que introducir dos veces, y pasar así la contraseña por el módulo de node.js que transforma el texto en un hash que se almacena posteriormente en la base de datos, en este caso actualiza la contraseña previamente guardada en el documento del usuario con la sesión iniciada.

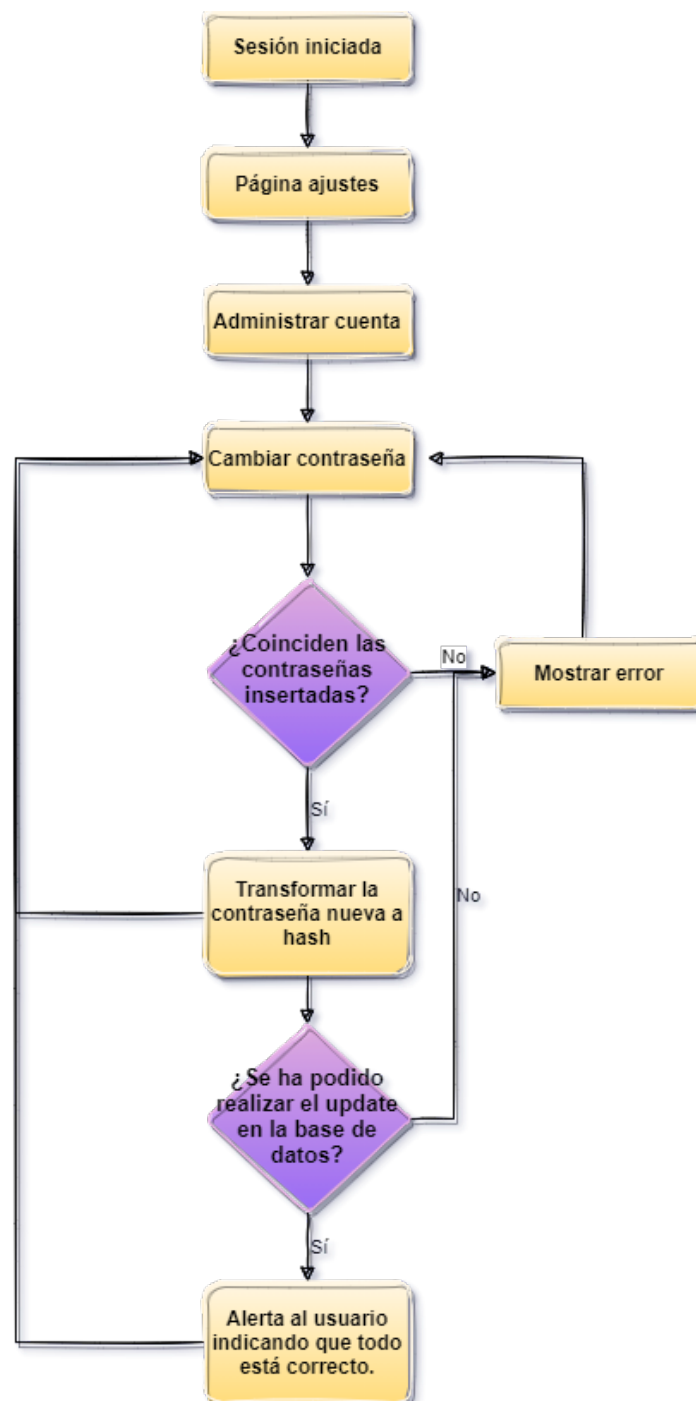


Figura 7.7: Diagrama del cambio de contraseña del usuario

## 7.2. Base de datos

A la hora de realizar el diseño de la base de datos, me he basado en los requisitos y escenarios previamente comentados. Pensando en los datos que la aplicación debía de almacenar y cómo debía de hacerlo. Debido a que los datos podrían ser un poco dispares, no podía utilizar una base de datos relacional al uso, por lo que me decanté por una base de datos no relacional, en este caso en MongoDB.

Por ello, la aplicación cuenta con una base de datos llamada *AppBeauty*, en la que cuenta con las diferentes colecciones que conforman mi proyecto. Son solamente tres, pero reúnen los datos estrictamente necesarios y con los que se realizan todas las funcionalidades del desarrollo.

En primer lugar, la colección de los usuarios, *Usuarios*, cuenta con los documentos que recogen los datos con los que se registran los usuarios. Los datos recogidos en esta colección son :

- **id\_user.** Un *id* que ayude a diferenciar a los usuarios.
- **name.** El nombre con el que se registra el usuario.
- **email.** El correo electrónico con el que se identifica al usuario.
- **password.** La contraseña para iniciar sesión.
- **date.** Fecha en la que se ha creado el usuario la cuenta.
- **zonaT.** Una de las zonas faciales a especificar por el usuario.
- **zonaMejillas.** Una de las zonas faciales a especificar por el usuario.
- **zonaBarbilla.** Una de las zonas faciales a especificar por el usuario.
- **activo.** El flag del borrado lógico, a *true* en caso de estar activo, *false* en caso contrario.

En segundo lugar, tenemos la colección de los ingredientes, *Ingredientes*, en la cual he recogido algunos de los ingredientes más famosos de los cosméticos faciales además de

añadir algunos a nivel general.

Los documentos tienen la siguiente forma:

- **INCI.** Es el sistema internacional para escribir los nombres de los ingredientes.
- **Traduccion.** La traducción al español del INCI del ingrediente.
- **PielAfectada.** Tipo de piel para la cual es nocivo.
- **EWG.** Número de referencia de toxicidad.
- **Web.** Link a la página desde la cual se ha sacado la información.

En tercer y último lugar, tenemos la colección de productos escaneados que el usuario ha decidido guardar y/o guardar como favoritos con el nombre que ha deseado, *Productos\_Escaneados*.

Los documentos de esta colección se recogen en los siguientes campos:

- **nombre.** Nombre con el cual ha querido guardar el usuario el producto.
- **ingredientes.** Lista de ingredientes que contiene ese producto escaneado.
- **id\_user.** El *id* del user que lo ha registrado.
- **favorito.** Flag que indica si el usuario ha marcado el producto como favorito o no, de tipo *booleano*.

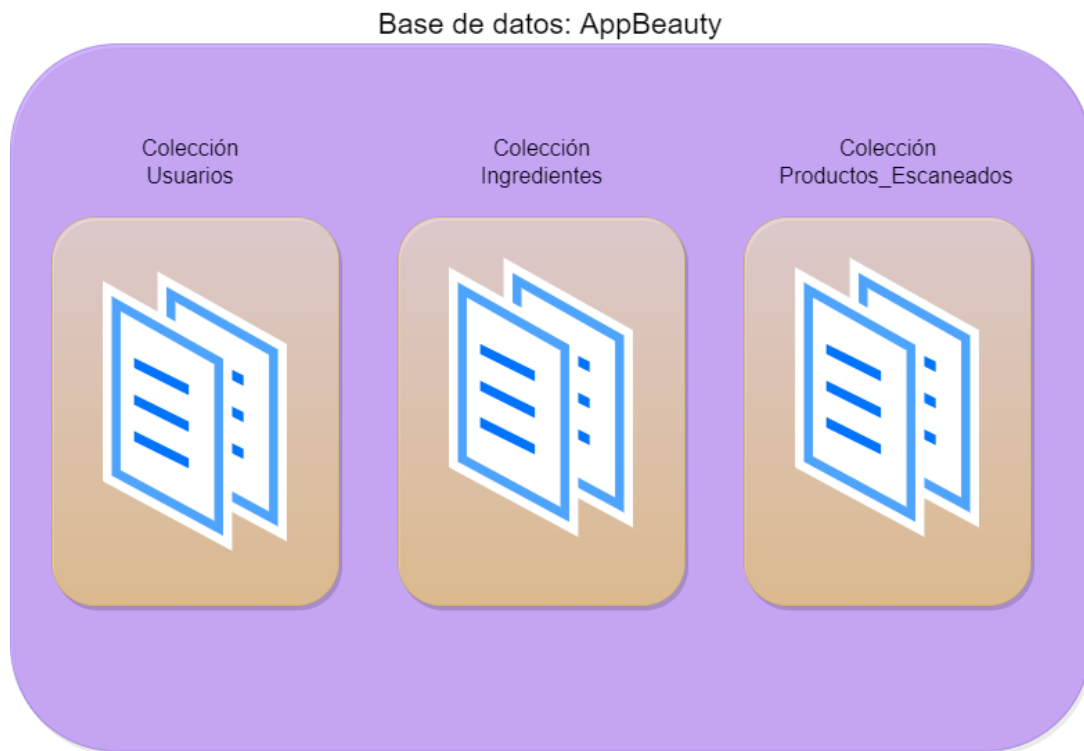


Figura 7.8: Diseño y colecciones de base de datos

## 7.3. Interfaz

En esta sección del diseño voy a mostrar capturas las funcionalidades de la aplicación mediante capturas de la edición final de la interfaz.

Cabe destacar que previo a esta edición final, realicé unos *mockups* a mano para diseñar toda la aplicación. El aspecto que tiene y las funcionalidades que quería mostrar.

### 7.3.1. Edición final

A continuación, voy a describir las interfaces de las cuales consta mi aplicación. En ellas se verán reflejadas las funcionalidades previamente explicadas para comprender así los escenarios y requisitos del capítulo 4.

La aplicación parte de una pantalla principal la cuál tiene acceso a las tres grandes secciones de la misma: ajustes, modificación del tipo de piel facial y por último las listas

de productos.

Seguidamente las describo detalladamente.

### Inicio de la aplicación

En esta primera pantalla, se puede ver información explicativa acerca de la aplicación, es fundamentalmente para usuarios no registrados y que no tienen acceso a las funcionalidades.

En ella se encuentran diferentes secciones en las que se explica el motivo de la aplicación y sus diferentes beneficios.

Además, tiene acceso directo al botón para registrarse o iniciar sesión.

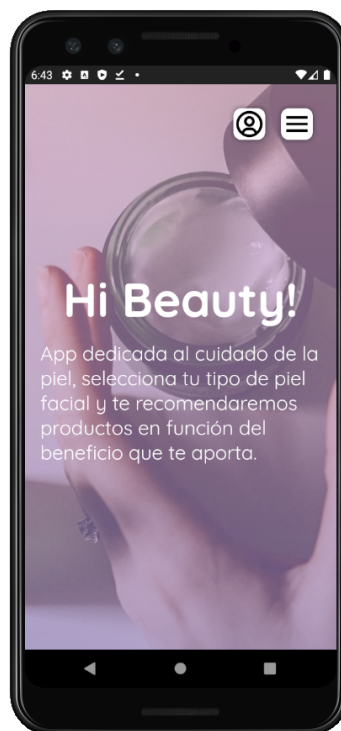


Figura 7.9: Interfaz de inicio

## Conexión con las funcionalidades

Una vez el usuario ha sido registrado o ha iniciado sesión en la aplicación, tiene acceso a esta pantalla en la cual puede navegar en las distintas funcionalidades que se ofrecen.

Puede acceder a las listas de sus productos, modificar y alterar el estado de las zonas faciales o realizar algún tipo de ajuste sobre su cuenta en la aplicación.



Figura 7.10: Interfaz principal

## Log in y sing up

En esta sección se pueden ver ambas vistas, en las cuales el usuario va a poder acceder por primera vez si realiza un registro o si por otro lado ya está registrado y lo que quiere hacer es iniciar sesión y disfrutar de la aplicación.

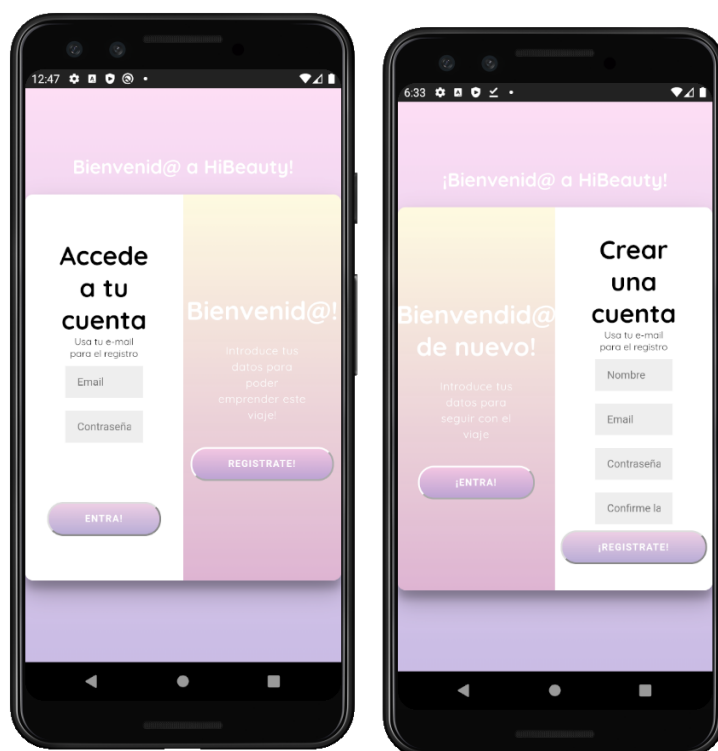


Figura 7.11: Interfaz del log in y del sing up

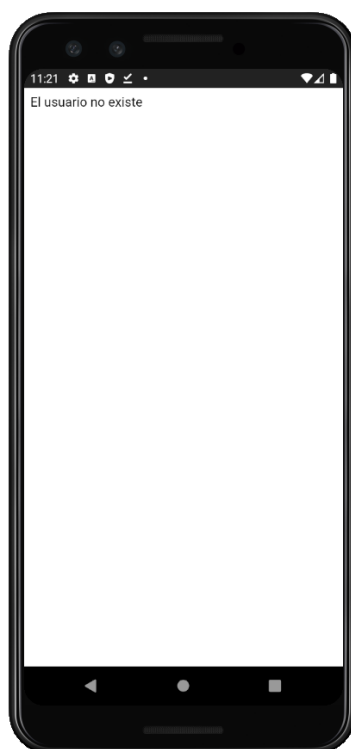


Figura 7.12: Interfaz de inicio de sesión fallido

Además, en caso de algún fallo, saldrá un mensaje como en la figura 7.12. En la cual el usuario tendrá que retroceder con el botón mismo del dispositivo móvil y aparecerá entonces de nuevo la pantalla de inicio de sesión, como en las figuras anteriormente nombradas, 7.11.

### Log out, mediante botón

En el caso de esta interfaz, el usuario tiene disponible esta posibilidad en cualquiera de las otras pantallas, ya que el botón es accesible desde cualquiera de éstas. Situado en la esquina superior derecha, junto con el nombre del usuario que esté utilizando en ese momento la aplicación.

El cierre de la sesión implica un borrado de las cookies generadas para almacenar la sesión del usuario. Si vuelve a algunas de las pantallas que requiere estar con la sesión iniciada, la aplicación no le va a permitir realizar ninguna de las acciones que se puedan llevar a cabo en las mismas.



Figura 7.13: Interfaz de cierre de sesión mediante botón superior



### Log out, mediante configuración

Otra manera en la que se puede realizar el cierre de la sesión en la aplicación es desde la interfaz de ajustes o configuración, tiene un botón de *Cerrar sesión* como se aprecia en la figura 7.14.



Figura 7.14: Interfaz de cierre de sesión mediante botón en configuración

## Modificación zona facial

La interfaz de la modificación o selección de la piel en función de la zona de la cara del usuario es una de las principales interfaces de la aplicación.

Consta como se ve en la figura 7.15, de unos radio buttons para elegir entre las tres opciones que se proponen y que intentan albergar de una manera general el tipo de piel de la mayoría de las personas: sensibles, grasa y seca.

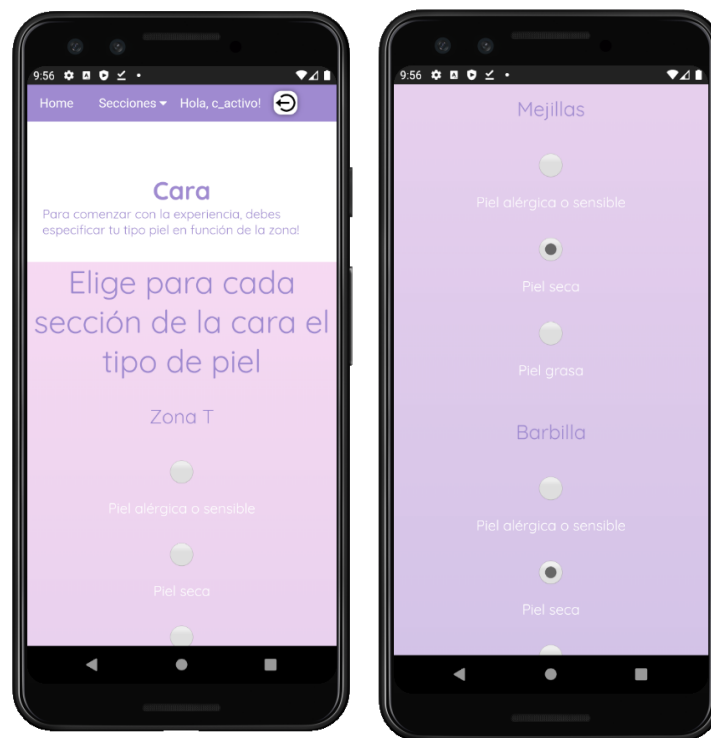


Figura 7.15: Interfaz modificación piel facial

Una vez el usuario ha decidido e investigado cuál es el tipo de piel que le corresponde, pulsa el botón de enviar, le aparecerá una alerta indicándole que se han modificado sus datos.

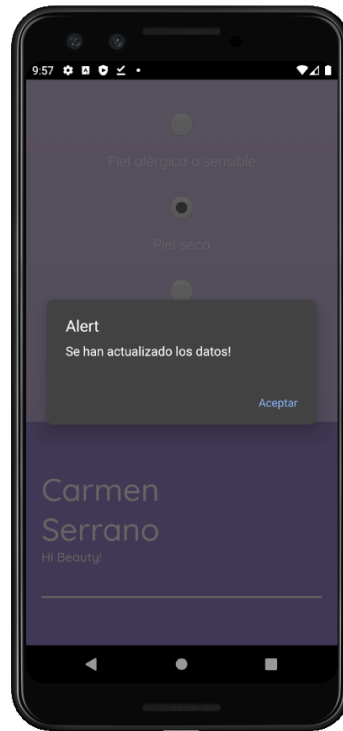


Figura 7.16: Alerta de confirmación

## Cámara

Esta interfaz de la cámara es otra de las principales de la aplicación. Es en la que el usuario va a poder ver la funcionalidad principal de ésta.

Consta de un botón con la que se puede hacer la recomendación del producto. Mediante una foto instantánea, con el único botón se acciona la cámara del dispositivo desde el cual el usuario va a poder realizar la foto al producto en ese mismo momento.

Antes de que el usuario realice la foto, se lanza una alerta para que el usuario sea consciente de que las recomendaciones basadas en la piel necesitan unos datos previos, y es por eso que se advierte que deben estar seleccionadas el tipo de piel facial antes de comenzar a sacar fotos, tal y como se observa en la figura 7.17.

En caso de que el usuario no haya seleccionado ningún tipo de piel facial, la aplicación no realizará ningún tipo de recomendación.

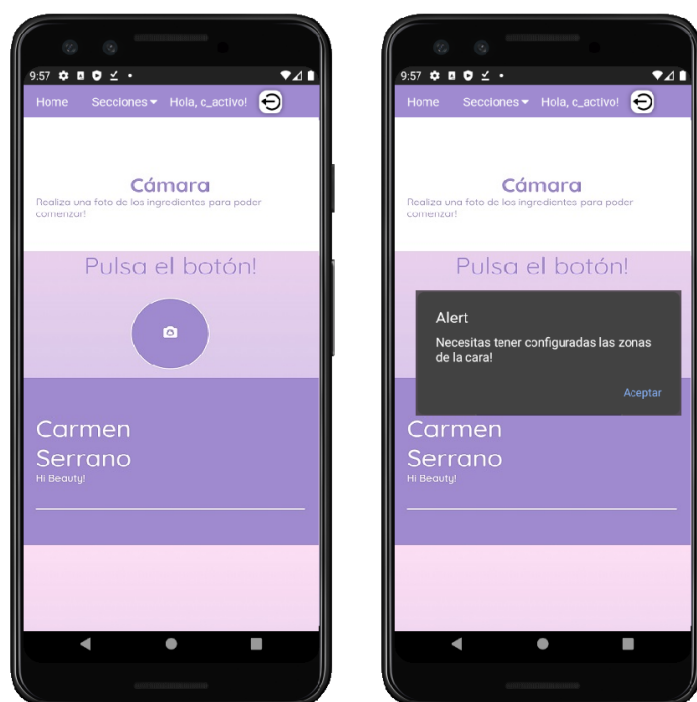


Figura 7.17: Interfaz de la cámara, alerta para el usuario de que tiene que tener configuradas las zonas de la cara

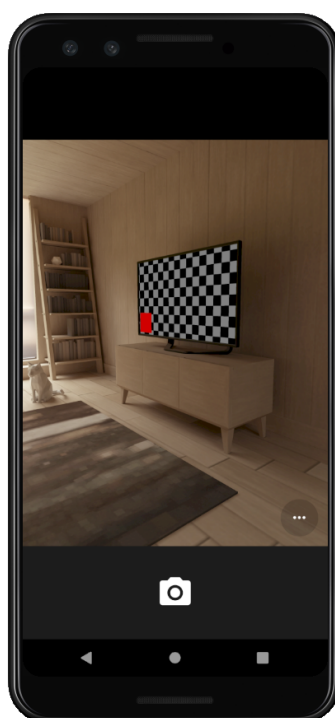


Figura 7.18: Cámara del emulador de Android

## Resultados

En esta interfaz, aparecerán dos tipos de respuesta, generados dinámicamente en función de los resultados obtenidos.

En el caso de que no haya ningún producto tóxico para el usuario aparecerá una imagen y un botón de vuelta hacia atrás que lleva al usuario de nuevo a la interfaz de la cámara.

En caso de que sí que haya algún producto nocivo para la piel previamente seleccionada por el usuario, aparecerá una lista de los ingredientes perjudiciales para el mismo, con un porcentaje de toxicidad y un color relacionado con ese porcentaje, para que así de una manera más visual el usuario pueda entender y decidir lo que quiere hacer o no con el producto recién escaneado.

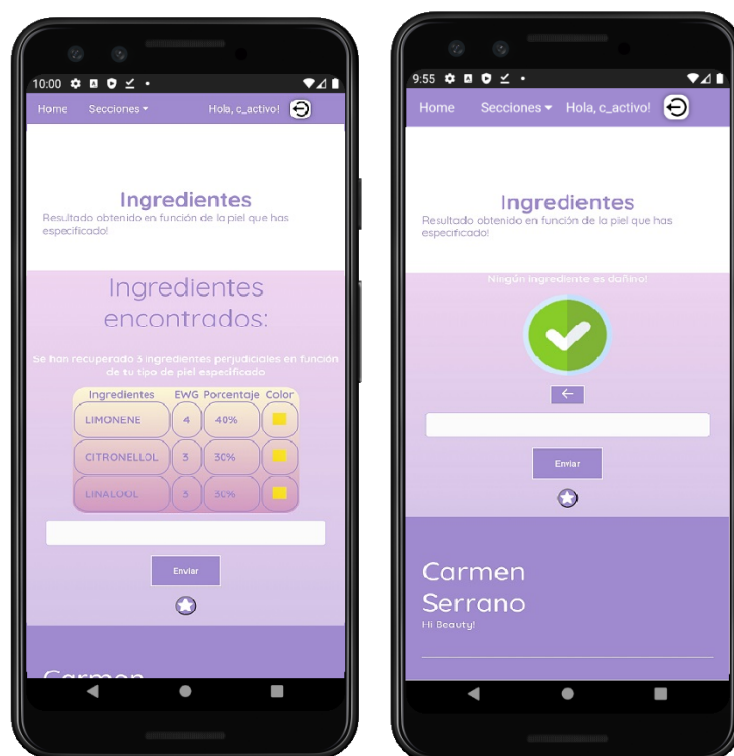


Figura 7.19: Interfaz de los resultados obtenidos

## Guardar producto

Para llegar a esta interfaz, se ha tenido que pasar previamente por la inmediatamente anterior, es decir, de la figura 7.17.

Ya que una vez escaneado el producto y mostrado los ingredientes tóxicos y que el usuario haya decidido qué hacer con dicho producto, se le ofrece la posibilidad de guardar dicho producto con un nombre seleccionado por el usuario para obtener así una mayor personalización de la aplicación, ya que cada usuario recordará el producto en función del nombre con el que lo ha decidido guardar.

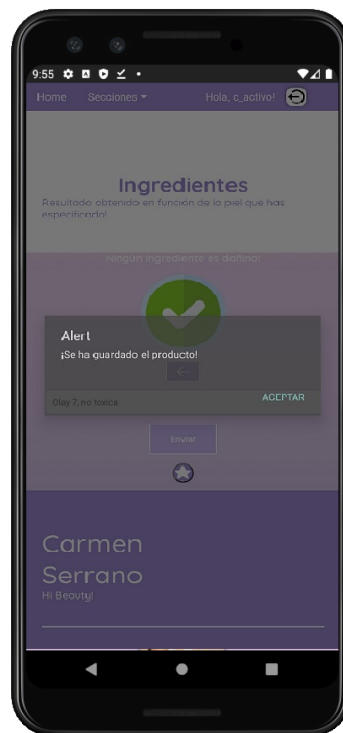


Figura 7.20: Interfaz para guardar producto

## Marcar favorito

Al igual que para guardar un producto, a la hora de guardar un favorito, tenemos que pasar por el mismo proceso que a la hora de guardarlo, como ya he indicado.

El usuario tiene que guardar previamente el producto o al menos introducir un nombre en el campo que se proporciona para ello. En ese entonces, en la base de datos se realiza un *update* en un flag en el que favorito pasa a tener un valor, para poder aparecer así en las listas de los productos favoritos del usuario.

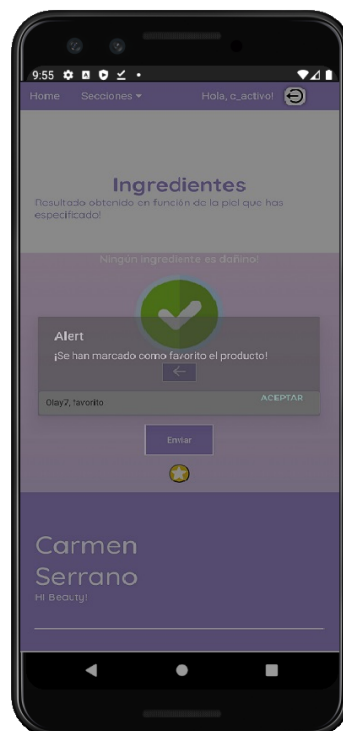


Figura 7.21: Interfaz de marcar como favorito un producto escaneado

## Visualizar listas guardados y favoritos

En el caso de esta interfaz, se accede desde la pantalla principal. La conexión entre algunas de las interfaces la explico visualmente en la figura 7.30.

Para poder visualizar las listas de los productos guardados y de los favoritos debe acceder a esta vista primero. Consta de dos botones uno para cada uno de las listas. Solamente se mostrarán productos para los usuarios que hayan decidido guardar éstos.

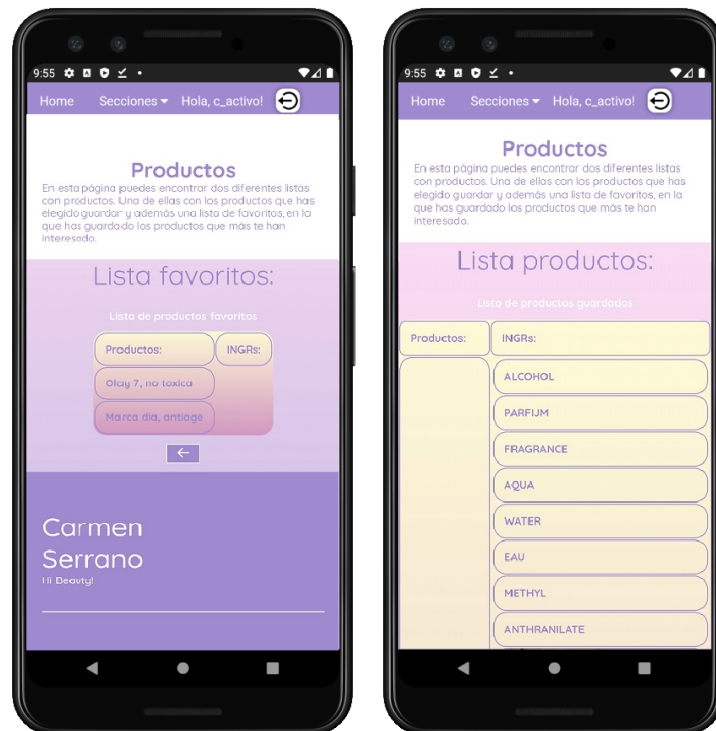


Figura 7.22: Interfaz que muestra ambas listas

## Ajustes

La interfaz de los ajustes cuenta con cuatro botones y cuatro vistas que se explican más adelante. El acceso a dicha interfaz, al igual que en la previamente explicada, el usuario debe de estar con la sesión iniciada y se accede desde la vista principal.

En esta sección el usuario puede realizar varias acciones que se comprenden desde realizar cambios en algunos de los aspectos de su perfil, así como notificar un problema si se ha encontrado un error en la aplicación o si hay algún otro tipo de error, hasta el cierre de



la sesión o incluso el borrado del perfil por completo de la aplicación.



Figura 7.23: Interfaz de los ajustes del usuario

## Administrar cuenta

En esta sección, el usuario podrá gestionar varios de los aspectos de la misma, como puede ser el cambio del nombre.

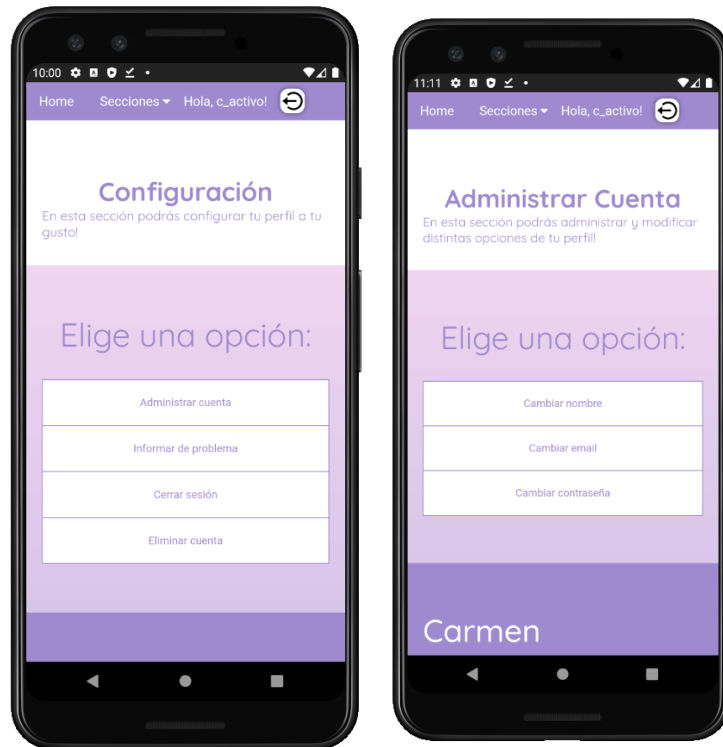


Figura 7.24: Interfaz configuración general e interfaz de la administración de la cuenta del usuario

- Cambiar nombre En esta interfaz el usuario dispone de un espacio para rellenar con el nuevo nombre que quiera obtener.

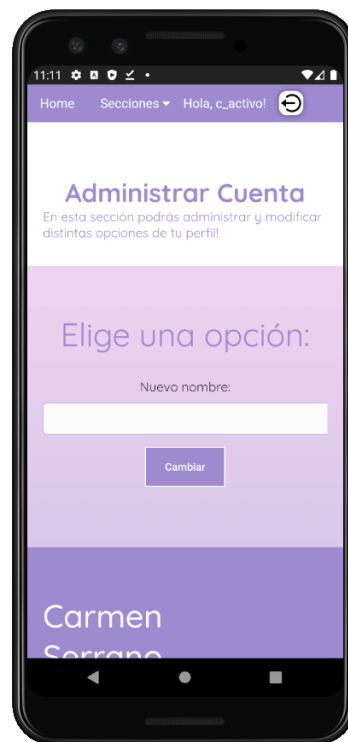


Figura 7.25: Interfaz de cambio de nombre

- Cambiar email Al igual que con el nombre, el usuario puede cambiar el email con el que se registró y con el que realizará el nuevo inicio de sesión.



Figura 7.26: Interfaz de cambio de correo electrónico

- **Cambiar contraseña** En caso de querer cambiar o modificar la contraseña, el usuario también cuenta con esa opción en esta interfaz, en la que además necesita introducir la contraseña dos veces al igual que a la hora de hacer el registro.s



Figura 7.27: Interfaz de cambio de contraseña

## Informar de un problema

En esta interfaz, el usuario cuenta con un formulario de contacto directo con el email de la aplicación, desde la que se gestionarían las peticiones de los usuarios o los errores encontrados por parte del mismo.

Cuenta con los campos de información del usuario así como un campo para la explicación del problema que hay o que se ha encontrado.



Figura 7.28: Interfaz de informar de un problema

## Eliminar cuenta

Esta interfaz no es una interfaz como tal porque el acceso a esta funcionalidad es el botón que se visualiza en la vista de los ajustes, previamente comentada.

El borrado de las cuentas de los usuarios es ficticia, ya que no permito que el usuario pueda realizar un borrado directamente en la base de datos y al no contar con un rol de gestión, he decidido hacer un borrado lógico. El borrado lógico se proporciona mediante un flag que indica si el usuario está o no está activo, en caso de no estar activo el usuario no podrá iniciar sesión en la aplicación y en caso de sí estarlo podrá acceder a ella.

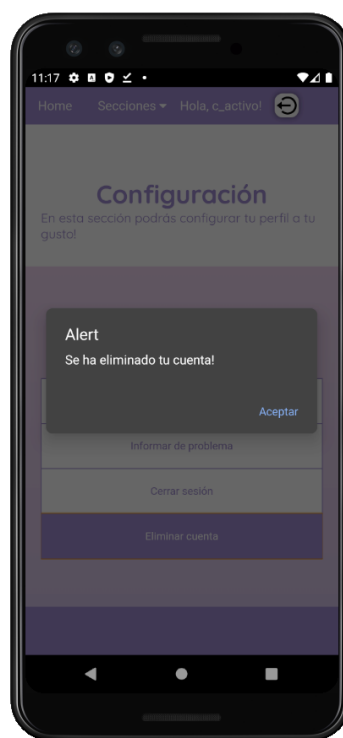


Figura 7.29: Interfaz para eliminar una cuenta de usuario

## Navegación entre interfaces

De cara a una mejor visualización a nivel global de la aplicación he creado un diagrama que comprenda todas las vistas y la relación que hay entre ellas. Como se puede observar, el acceso a la aplicación en sí sólo se puede obtener mediante el inicio de sesión o registro previo del usuario, en caso de no hacerlo sólo se puede acceder a la página principal en la cuál aparece la explicación y descripción de este proyecto.

Como núcleo central nos encontramos con la página principal, desde la cual se puede acceder al resto de vistas que comprenden este proyecto junto a sus funcionalidades. Las cuatro principales que son: mostrar las listas de los productos, editar las zonas faciales del usuario, el acceso a la cámara para el escaneo de los productos y por último los ajustes de la cuenta.

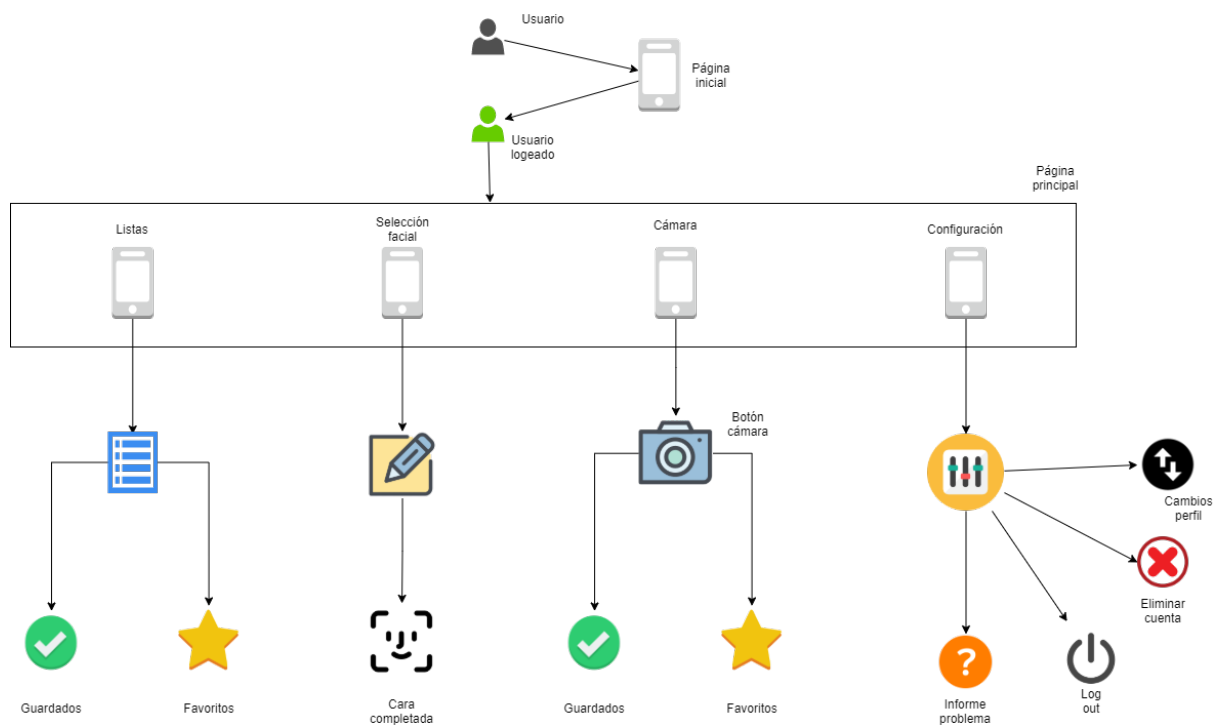


Figura 7.30: Diagrama conexión entre interfaces



## 7.4. Conclusiones del capítulo

Finalmente, todas las interfaces han sido desarrolladas desde un punto de vista en el cual se facilite el uso por parte del usuario de la aplicación. He querido un diseño y un acceso a las funcionalidades básico e intuitivo pero a la vez estético.

Todas las vistas están relacionadas y puedes acceder a todas ellas desde a su vez, todas ellas desde la pestalla de secciones que hay en la barra superior de la aplicación, en caso de que el usuario se confunda o que sea más fácil el acceso para él sin tener que pulsar el botón de regreso del dispositivo móvil del usuario.

# Capítulo 8

## Tecnologías

### 8.1. Tecnologías

#### 8.1.1. Node.js

Node.js es un framework que utiliza JavaScript y que además es de código abierto. Se usa fundamentalmente para el desarrollo de la capa del servidor, que es para lo que yo lo he utilizado.



Es la principal opción elegida por muchos desarrolladores por múltiples características y entre ellas cabe destacar las cuales por las que yo lo he seleccionado.

Figura 8.1: OpenVPN

Es asíncrono y controlado por eventos, todas las API de la biblioteca Node.js son asíncronas por lo que no se pueden bloquear, por lo que el servidor que yo he desarrollado nunca espera que la API devuelva los datos.

Además, está construido en el motor de JavaScript V8 de Google Chroma, con lo que es muy rápida a la hora de la ejecución del código.

Tiene licencia basada en código abierto, por lo que se distribuye bajo una licencia MIT.

## **Bcryptjs**

Este módulo, permite la utilización de algoritmos hash que convierten cualquier cantidad de datos en éste de una longitud fija que no se puede revertir.

Es una librería de encriptación que me ha ayudado a cifrar las contraseñas de los usuarios para guardar de una manera segura las mismas.

El módulo se utiliza cuando el usuario se registra y de ahí en adelante, si el usuario inicia sesión se comprueba el hash generado de la contraseña que teclea éste con el hash almacenado en base de datos, si coincide tendrá acceso a la aplicación y si el hash generado no coincide con el almacenado no se podrá realizar el login.

## **Express**

Es un marco de aplicación web para Node.js. Proporciona varias características que hacen que el desarrollo de las aplicaciones web, como es en este caso la mía, sea más rápido y fácil.

Express, se basa en un módulo middleware de Node.js llamado connect, que utiliza a su vez el módulo http.

Lo he utilizado para poder definir rutas de mi aplicación basadas en métodos POST y GET.

Además, es el módulo con la conexión más fácil con MongoDB, lo cual agilizó bastante mi trabajo.

## **Mongoose**

Mongoose es un Object Document Mapper (ODM) que permite definir los objetos utilizados en la aplicación.

Estos objetos han sido previamente creados y almacenados en la base de datos de MongoDB.

Gracias a la funcionalidad de moongoose, se pueden crear los esquemas necesarios para la bbdd, así como el acceso a los mismos desde el servidor.

### **Ocr-space-api-wrapper**

Esta tecnología es la que he utilizado para el reconocimiento de imágenes.

Es un paquete específico para node, para poder utilizar dicha tecnología. El OCR es el reconocimiento óptico de caracteres que he explicado en capítulos anteriores. El objetivo principal es obtener el texto de una imagen.

Este módulo lo he utilizado para obtener los ingredientes de la foto que decide el usuario subir. Es una tecnología que pertenece a OCR.Space, que fue la seleccionada por el plan gratuito que ofrecía, ya que tiene buenas conexiones simultáneas, que no he llegado a gastar y que para el funcionamiento actual de la aplicación es más que suficiente.

### **Connect-flash**

Es un paquete para Express, como comento en la parte superior de esta sección. Permite mostrar mensajes en la pantalla bajo ciertas condiciones.

Éste módulo lo he utilizado para avisar al usuario cuando finalizaba una funcionalidad como por ejemplo el registro de un nuevo producto, si ha marcado como favorito dicho producto o si ha actualizado la zonas de la cara.

### **Cookie-parser**

Cookie-parser es un módulo que ayuda con el manejo de cookies como su propio nombre indica. Es necesario tener el módulo de Express.js porque se apoya en él para su funcionalidad.

Nos ayuda con la configuración de las cookies que se utilizan en la aplicación.

En mi caso, he recurrido a este módulo porque necesitaba controlar algunos datos de la sesión del usuario para algunas de las funcionalidades, como que se pueda realizar el guardado de las zonas de la cara que selecciona el usuario, o de que se pueda realizar el

correcto cierre de la sesión del usuario mismo que la ha empezado.

## Passport

Este framework, depende al igual que el anterior de Express.js yq ue se complementa con el ya comentado Connect.

Éste apoya a la aplicación dotandolo de un servicio gratuito para gestionar la autenticación.

Cabe destacar que el framework cuenta con la autenticación desde diferentes opciones como desde redes sociales.

### 8.1.2. MongoDB

Esta base de datos está orientada a documentos, lo que la diferencia claramente de las usualmente conocidas bases relacionales. Esto significa que los datos se guardan en documentos en vez de en registros. Éstos se almacenan en BSON, que es una representación binaria de JSON.



Figura 8.2: MongoDB

La diferencia más clara con las bases de datos relacionales es que ésta no sigue ningún esquema. Los documentos de una misma colección pueden tener diferentes campos, más o menos y almacenar todo tipo de objetos.

Las única dificultad que le he encontrado es a la hora de relacionar las tablas, ya que no se puede. Igualmente he optado por este tipo de base de datos porque los datos que iba a obtener podían ser dispares y una base de datos relacional era demasiado estricta para mi aplicación.

MongoDB aporta facilidad al proyecto y flexibilidad a la hora de almacenar y acceder a los datos obtenidos o los generados por el usuario.

### 8.1.3. Latex

Latex es un sistema de preparación de documentos con el que se pueden redactar diferentes tipos de escritos, ya sean artículos, tesis o documentos.

The LaTeX logo, which consists of the word "LATEX" in a stylized, serif font. The letters are black and have a slightly irregular, hand-drawn appearance.

Figura 8.3: Latex

Latex ayuda con la escritura de fórmulas matemáticas o químicas y en general está relacionado con la escritura de documentos científicos.

### 8.1.4. OpenVPN

La herramienta de OpenVPN, es una herramienta de conectividad basada en software libre. Ofrece una conexión de punto a punto que valida dicha conexión entre usuarios y hosts conectados de manera remota. Para el proyecto, he decidido utilizarla para poder realizar una conexión segura con el servidor.



Figura 8.4: OpenVPN

## 8.2. Conclusiones del capítulo

En conclusión, las tecnologías utilizadas para realizar el proyecto se decidieron en base a las necesidades del mismo, intentando seleccionar las que más se adecuaban a mis conocimientos obtenidos en la carrera.

Además, he seleccionado algunas tecnologías con las que no estaba familiarizada para obtener nuevas capacidades a la hora de desarrollar la aplicación, lo cual ha tomado tiempo de investigación y documentación por mi parte para aprender a tiempo y poder aplicar estos nuevos conocimientos y volcarlos en la misma.

# Capítulo 9

## Conclusiones

### 9.1. Conclusiones

Actualmente, como ya he comentado con anterioridad, hay un auge en cuanto al cuidado de la piel se refiere. En estos momentos, el cuidado personal es un tema a la orden del día, se ven en los telediarios y sobretodo en las redes sociales, lo que crea una tendencia en las personas de preocupación ya sea bien por su piel o su alimentación.

En base a esa necesidad generada se me ocurrió esta idea, considerando que puede haber usuarios interesados en cuidar de una manera específica la piel facial, ya que es la más sensible y la más expuesta.

Para poder llevar a cabo la idea, se ha desarrollado esta aplicación web-apk. Que consta de tres funcionalidades básicas para cumplir con la finalidad que expongo más arriba.

La parte fundamental del proyecto consta de un sistema de recomendación, en función de dos factores muy específicos: uno ellos es la fotografía tomada por parte del usuario a los ingredientes del producto de interés y el segundo es el tipo de piel que haya especificado el usuario en la aplicación. Dando esto como resultado a una lista de ingredientes dañinos, en caso de que existan y que puedan perjudicar así a la piel del usuario. Lo que le ayudará a decidir si quiere o no comprar o seguir utilizando el producto escaneado.

Las otras dos funcionalidades consisten en mostrar diferentes listas que comprenden los productos que el usuario ha decidido almacenar o no, y también, la posibilidad de configurar la cuenta del usuario, administrando la información que tiene dentro de la aplicación

La implementación de este proyecto se ha llevado a cabo apoyandome en una aplicación llamada Cordova, la cual ha facilitado el trabajo ya que gracias a ella se puede tener dos versiones de este proyecto, una para utilizarlo desde la página web o mediante una apk para el dispositivo móvil, reuniéndolo todo bajo un mismo código.

La estructura en la que se ha basado la aplicación, he intentado que sea clara y que esté bien seccionada de cara a trabajos futuros sobre la misma en caso de continuar con el desarrollo, además de facilitarme la visión global del proyecto.

Considero que el alcance del desarrollo es proporcional al tiempo en el que se ha realizado, no obstante con más tiempo y más medios considero que podría pasar de ser un concepto a un desarrollo más profesional.

Para finalizar quiero destacar que estoy contenta con el transcurso, el proceso y el resultado de este proyecto.

## 9.2. Conclusions

Currently, as I have mentioned previously, there is a boom in terms of skin care. At this time, personal care is a topic on the order of the day, they are seen on the news and especially on social networks, which creates a trend in people who are concerned either about their skin or their diet.

Based on this generated need, I came up with this idea, considering that there may be users interested in to be more attentive to skin care, since it is the most sensitive and the most exposed.

In order to carry out the idea, this web-apk application has been developed. That consists of three basic functionalities to fulfill the purpose that I expose above.



The fundamental part of the project consists of a recommendation system, based on two very specific factors: one is the photograph taken by the user of the ingredients of the product of interest and the second is the type of skin specified by the user in the app. Resulting in a list of noxious ingredients, if they exist and may harm the user's skin. This will help that user decide if they want buy the product or stop using it.

The other two functionalities consist of showing different lists that include the products that the user has decided to store or not, and also, the possibility of configuring the user's account, managing the information they have within the application

The implementation of this project has been carried out with the support of an application called Cordova, which has facilitated the work since thanks to it you can have two versions of this project, one to use it from the web page or through an apk in a mobile device, and all of that with just one code.

The structure on which the application has been based, I have tried to be clear and well sectioned for future work on it in case of continuing with the development, in addition to providing me with the global vision of the project.

I believe that the scope of the development is proportional to the time in which it has been carried out, however with more time and more means I believe that it could go from being a concept to a more professional development.

Finally, I want to emphasize that I am happy with the course, the process and the result of this project.

# Capítulo 10

## Trabajo futuro

### 10.1. Trabajo futuro

En esta sección voy a indicar algunas funcionalidades e ideas que considero que podrían enriquecer el proyecto pero que por tiempo ha sido poco posible llevarlas a cabo.

- **Rutina facial**, para poder aportar más a la aplicación siguiendo en la línea de cuidado de la piel, una sección en la cuál el usuario pudiera registrar su rutina facial, con productos que haya escaneado previamente o no, pudiendose almacenar otro tipo de productos y de distinta manera, al alcance del usuario. Ésta consistiría en otra vista principal del proyecto en la cual el usuario podría incluso administrar la aplicación para que ésta genere notificaciones en el dispositivo a modo de recordatorio para el usuario.

Además se mostraría la lista de productos por orden en el que el usuario quiera situarlos y que pueda ir tachando los que haya utilizado ese día o simplemente que se puedan tachar según estás haciendo la rutina, para así tener completado un día más de la rutina facial.

Para realizar este trabajo, tendría que añadir una vista más a la página principal en

la cual primero el usuario registre los productos con nombre, los cuales tendría que almacenar en una colección aparte dedicada exclusivamente a esta funcionalidad en la que guardaría todos los productos y el orden en el que el usuario los querría utilizar y el *id* del usuario.

- **Piel corporal**, ya que es una aplicación dedicada al cuidado de la piel facial, considero que aunque la piel corporal puede ser más heterogénea, también se pueden hacer recomendaciones para ésta. Por tanto, el usuario podría obtener resultados de ingredientes dañinos pero de productos dedicados al cuidado de la piel corporal, en vez de para la facial.

Para ello tendría que almacenar en el usuario otro tipo de dato que indicara zonas del cuerpo para que el usuario pudiera configurarlo, así como realizar cambios en el momento de escanear la foto, para que el usuario indique si el producto es de cuidado facial o si es para el cuidado corporal, de este modo la recomendación sería diferente ya que hay productos con ingredientes que son perjudiciales a nivel facial pero no para el cuerpo. Con esto también se tendría que añadir un campo más en la base de datos, en la colección de los ingredientes para poder diferenciar entre el daño que produce a la cara y al daño que produce en la piel del cuerpo, cambiando así algunas de las consultas para las recomendaciones de productos.

- **Guía recomendación**, ya que esta aplicación se ha diseñado para el cuidado facial y se da por hecho que el usuario sabe qué tipo de piel tiene en qué zona de la cara específica, puede ser que el usuario no cuente todavía con esa información, por lo que me parecería útil y enriquecedor el poner una guía para ayudar al usuario a distinguir el tipo de piel que tiene para poder usar así la aplicación.

Sería una vista meramente informativa con texto obtenido de varias fuentes contrastadas para ayudar al usuario.

- **Foto de perfil**, para que el usuario puede tener aún más personalizado su perfil y pueda elegir una foto de perfil que suba desde su cámara del móvil.

# Índice de figuras

1.1. Plan de trabajo en Trello . . . . .	4
2.1. Working plan on Trello . . . . .	3
4.1. Foto encuesta realizada . . . . .	12
6.1. Diagrama con la arquitectura del proyecto . . . . .	23
7.1. Diagrama de inicio de sesión . . . . .	29
7.2. Diagrama de escaneo de una fotografía . . . . .	30
7.3. Diagrama de selección zonas faciales . . . . .	31
7.4. Diagrama de cómo guardar un producto previamente escaneado . . . . .	32
7.5. Diagrama de cómo guardar un producto previamente escaneado . . . . .	33
7.6. Diagrama de borrado lógico de la cuenta del usuario . . . . .	34
7.7. Diagrama del cambio de contraseña del usuario . . . . .	35
7.8. Diseño y colecciones de base de datos . . . . .	38
7.9. Interfaz de inicio . . . . .	39
7.10. Interfaz principal . . . . .	40
7.11. Interfaz del log in y del sing up . . . . .	41
7.12. Interfaz de inicio de sesión fallido . . . . .	41
7.13. Interfaz de cierre de sesión mediante botón superior . . . . .	42
7.14. Interfaz de cierre de sesión mediante botón en configuración . . . . .	43
7.15. Interfaz modificación piel facial . . . . .	44
7.16. Alerta de confirmación . . . . .	45
7.17. Interfaz de la cámara, alerta para el usuario de que tiene que tener confi- guradas las zonas de la cara . . . . .	46

---

7.18. Cámara del emulador de Android . . . . .	46
7.19. Interfaz de los resultados obtenidos . . . . .	47
7.20. Interfaz para guardar producto . . . . .	48
7.21. Interfaz de marcar como favorito un producto escaneado . . . . .	49
7.22. Interfaz que muestra ambas listas . . . . .	50
7.23. Interfaz de los ajustes del usuario . . . . .	51
7.24. Interfaz configuración general e interfaz de la administración de la cuenta del usuario . . . . .	52
7.25. Interfaz de cambio de nombre . . . . .	53
7.26. Interfaz de cambio de correo electrónico . . . . .	54
7.27. Interfaz de cambio de contraseña . . . . .	55
7.28. Interfaz de informar de un problema . . . . .	56
7.29. Interfaz para eliminar una cuenta de usuario . . . . .	57
7.30. Diagrama conexión entre interfaces . . . . .	58
8.1. OpenVPN . . . . .	60
8.2. MongoDB . . . . .	63
8.3. Latex . . . . .	64
8.4. OpenVPN . . . . .	64

# Bibliografía

- [1] Node.js - Documentación oficial <https://nodejs.org/es/docs/>
- [2] Express.js - Documentación oficial <https://expressjs.com/es/>
- [3] Mongoose - Documentación oficial <https://mongoosejs.com/docs/>
- [4] mongo Shell Methods - Documentación oficial <https://docs.mongodb.com/manual/reference/method/>
- [5] Queries - Documentación oficial <https://mongoosejs.com/docs/queries.html>
- [6] Cordova - Documentación oficial <https://cordova.apache.org/docs/es/latest/guide/overview/>
- [7] Lista de la UE de productos prohibidos en la UE. [https://ec.europa.eu/growth/tools-databases/cosing/index.cfm?fuseaction=ref\\_data.annexes\\_v2/](https://ec.europa.eu/growth/tools-databases/cosing/index.cfm?fuseaction=ref_data.annexes_v2/)
- [8] Artículo que indica los ingredientes peligrosos en función del tipo de piel. [https://www.vanitatis.elconfidencial.com/estilo/belleza/2020-06-16/ingredientes-piel-mixta-seca-grasa-formulas\\_2629947/](https://www.vanitatis.elconfidencial.com/estilo/belleza/2020-06-16/ingredientes-piel-mixta-seca-grasa-formulas_2629947/)
- [9] Biblioteca ordenada alfabéticamente de los ingredientes en español. <https://beautypul.com/2018/09/guia-de-ingredientes-cosmeticos/>
- [10] OCRSpace - Documentación oficial <https://ocr.space/ocrapi>
- [11] ocr-space-api-wrapper - Documentación oficial <https://www.npmjs.com/package/ocr-space-api-wrapper>
- [12] Ejemplos de aplicaciones parecidas <https://dominaturosacea.com/mejores-apps-para-analizar-ingredientes-cosmeticos/>

- 
- [13] Ejemplos e información de Skin360 <https://mujermexico.com/moda-y-belleza/5-apps-que-te-ayudan-a-cuidar-la-piel-de-tu-rostro/>
- [14] Ejemplos e información de Skin360 <https://mujermexico.com/moda-y-belleza/5-apps-que-te-ayudan-a-cuidar-la-piel-de-tu-rostro/>
- [15] Ejemplos e información de SkinSmart <https://novataenprogreso.blogspot.com/2019/04/tips-aplicaciones-gratuitas-que-te.html>
- [16] Información de Google Vision con explicación externa <https://medium.com/@sebasgojs/introducci%C3%B3n-a-google-vision-api-2103161166e9>
- [17] Información de Google Vision Oficial <https://cloud.google.com/vision/pricing?hl=es>
- [18] Información de Azure Vision Oficial <https://docs.microsoft.com/es-es/azure/cognitive-services/computer-vision/concept-recognizing-text>
- [19] Información sobre el término OCR <https://debitoor.es/glosario/ocr-escaneo>
- [20] Información acerca de la importancia del cuidado de la piel <https://likabio.com/cuidado-la-piel-importante/#:~:text=La%20importancia%20de%20cuidar%20la%20pieltext=Nos%20protege%20>
- [21] Estudio que explica auge del consumo de productos cosméticos <https://www.stanpa.com/notas-prensa/el-consumo-de-perfumes-y-cosmeticos-crece-por-cuarto-ano-consecutivo-hasta-los-6-954-millones-e/>
- [22] Gráficos de consumo en productos cosméticos por país <https://es.statista.com/grafico/10676/los-paises-mas-coquetos/>
- [23] Fotografía encuesta por cookie *studio* <https://www.freepik.es/fotos/mano>
- [24] Información sobre conexión cámara de dispositivo a aplicación <https://www.adictosaltrabajo.com/2014/01/16/acceso-camara-phonegap/>
- [25] Información sobre conexión cámara de dispositivo a aplicación 2 <https://ourcodeworld.co/articulos/leer/334/como-crear-una-aplicacion-movil-hibrida-para-android-con-cordova-en-windows-desde-cero>
-



- [26] Fotografías utilizadas para el desarrollo de la aplicación, de uso gratuito.  
<https://www.pexels.com/es-es/@shiny-diamond>
- [27] Información ingredientes tóxicos, artículo <https://www.hola.com/belleza/caraycuerpo/2019121015402/cosmetica-evitar-tipo-piel/#:~:text=Las%20pieles%20grasas%20se%20beneficiar%C3%A1n,funci%C3%A9n>
- [28] Información sobre productos tóxicos y su legislación en la Unión Europea  
<https://chemicalsinourlife.echa.europa.eu/es/chemicals-in-cosmetics>
- [29] Podcast que trata sobre los ingredientes y productos cosméticos y su tratamiento y legislación en la Unión Europea <https://podtail.com/no/podcast/revolucion-sostenible/-1-laura-bey-mitos-y-realidades-de-la-industria-co/>
- [30] Información etiquetado en EEUU, nomenclatura INCI. <https://www.elsevier.es/es-revista-offarm-4-articulo-productos-cosmeticos-13065404>
- [31] Regulación e información sobre el etiquetado en la Unión Europea por Stanpa <https://www.stanpa.com/innovacion-y-ciencia/cosmeticos-regulacion/transparencia-etiquetado/>
- [32] Metodología Evaluación INCI [https://inci.es/es/blog/metodologia\\_evaluacion](https://inci.es/es/blog/metodologia_evaluacion)
- [33] Legislación sobre productos cosméticos por el gobierno de España  
<https://www.aemps.gob.es/cosmeticos-cuidado-personal/legislacion/cosmeticos/>
- [34] Ingredientes más utilizados según INCI, utilizados para rellenar mi base de datos.  
<https://incibeauty.com/es/ingredients>
- [35] Artículo que habla sobre las reacciones alérgicas de las pieles sensibles con los perfumes  
<https://www.vogue.es/belleza/articulos/cosmeticos-sin-perfumes-fragancias-cuidado-piel-sensible-reactiva>
- [36] Ingredientes tóxicos en cosméticos, artículo <https://www.cremas-caseras.es/blog/ingredientes-toxicos-en-cosmeticos/>
- [37] Artículo que trata sobre la piel sensible y los productos tóxicos <https://organics-magazine.com/piel-sensible-atopica-o-reactiva-iii-mas-quimicos-nocivos/>
- [38] Evaluación seguida por EWG, utilizada en el TFG  
<https://www.ewg.org/skindeep/contents/users-guide/>

- 
- [39] Principales ingredientes tóxicos a evitar <https://www.amapolabio.com/cultiva-tu-piel/ingredientes-toxicos-que-deberias-evitar-en-tus-cosmeticos/>
- [40] Código utilizado de apoyo para construir el formulario <https://javiniguez.com/crea-un-formulario-de-contacto-en-html-y-css/>
- [41] Página para obtener iconos utilizados en la aplicación <https://boxicons.com/>
- [42] Página para obtener imágenes utilizadas en la aplicación <https://www.manypixels.co/gallery>
- [43] Página para obtener gradientes de color utilizados en la aplicación <https://webgradients.com/>
- [44] Página para obtener imágenes utilizados en la aplicación <https://unsplash.com/>

This work is licensed under a Creative Commons  
“Attribution-NonCommercial-NoDerivatives 4.0 International” license.

